

网络安全控制机制

林 闯 蒋屹新 尹 浩 著

清华大学出版社

网络安全控制机制

林　闯　蒋屹新　尹　浩　著

清华大学出版社
北　京

内 容 简 介

网络安全是计算机和通信领域很重要的研究方向，而网络安全控制机制是网络安全的基本保障，是网络安全中的重要研究内容。本书分为5章，第1章是访问控制机制，讲述了访问控制的最新进展，并讨论了移动通信和可信网络环境下的访问控制技术。第2章是认证机制，介绍AAA服务器的认证原理及其在无线网络中的应用，然后介绍多级安全域的认证模型，最后讨论了移动网络中的可以容忍DoS攻击的认证模型。第3章是数字签名机制，介绍数字签名中的公钥密码体制和椭圆曲线密码体制，并讨论了基于椭圆曲线的群体导向的签名方案。第4章是密钥管理机制，概述了基本的组密钥分发机制，讨论了自愈的组密钥分发协议和基于时限的组密钥分发机制，并阐述了无线传感器网络中的密钥管理。第5章是基于应用层组播的视频安全机制，介绍流媒体与应用层组播，数字水印技术以及视频加密技术，并详细描述了一个视频安全组播协议，讨论了视频流传输过程中的差错控制。

本书全面、系统地展示了网络安全控制机制的研究内容和最新成果，具有完整性、实用性和学术性。非常适合我国计算机网络和通信领域的教学、科研工作和工程应用参考。既可以供计算机、通信、电子、信息等相关专业的研究生和大学高年级学生作为教材或教学参考书，也可以供计算机网络研究开发人员、网络运营商等网络工程技术人员参考。

图书在版编目（CIP）数据

网络安全控制机制/林闯，蒋屹新，尹浩著．—北京：清华大学出版社，2008.12

ISBN 978-7-302-18673-1

Ⅰ．网…　Ⅱ．①林…②蒋…③尹…　Ⅲ．计算机网络—安全技术　Ⅳ．TP393.08

中国版本图书馆CIP数据核字(2008)第152079号

责任编辑：薛　慧

责任校对：赵丽敏

责任印制：

出版发行：清华大学出版社　　**地　址**：北京清华大学学研大厦A座

http://www.tup.com.cn　　**邮　编**：100084

社 总 机：010-62770175　　**邮　购**：010-62786544

投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn

质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn

印 刷 者：

装 订 者：

经　销：全国新华书店

开　本：185×260　**印　张**：20.25　**字　数**：485千字

版　次：2008年12月第1版　**印　次**：2008年12月第1次印刷

印　数：1～0000

定　价：0.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题，请与清华大学出版社出版部联系调换。联系电话：010-62770177转3103　　产品编号：-

FOREWORD

前　言

背景

随着传感器、嵌入式设备、消费电子等设施的大量接入，互联网络在规模和应用领域上日益得到拓展，网络的规模仍在继续扩大，网络在国民经济生活中的基础性和全局性作用日益增强。尽管互联网已经转变并大大改善了人类社会的经济和生活方式，但同时也不得不面临大量的网络安全问题，如恶意攻击、垃圾邮件、计算机病毒、不健康资讯等。尽管信息网络的安全研究已经持续多年，但对网络攻击和破坏行为的对抗效果并不理想，仍然面临着严峻的挑战。

网络安全是计算机和通信领域重要的研究内容，而对网络安全控制机制的研究是保障网络安全的基本技术。国际标准化组织（ISO）在网络安全标准 ISO 7498—2 中定义了 5 种层次型安全服务：身份认证服务、访问控制服务、数据保密服务、数据完整性服务和不可否认服务。其中，访问控制是信息安全的一个重要组成部分，它作为系统安全的关键技术，既是一个老生常谈的内容又面临着新的挑战。随着网络技术的发展，访问控制技术也将作为网络安全的一个重要方面日益受到更多人的关注。授权和认证是访问控制的基础，正确的授权实际上依赖于认证。如何保证用户身份的真实性和合法性，如何正确授权用户的权限，是访问控制和认证机制中重要的研究内容。

虽然使用数据加密、访问控制等多项技术可以对数据通信时的保密性和完整性予以保证，然而仅仅有这些还是不够的，特别是近年来，随着电子商务的发展，人们通过通信网络进行迅速的、远距离的贸易，数字或电子签名也应运而生，并开始用于商业通信系统。这些都要求根据不同的情况设计出适合特定情况的安全而有效的数字签名，以适应飞速发展的网络环境下的安全需要。因此，数字签名也是网络安全机制中一项重要内容，其中基于椭圆曲线的数字签名方案成为热点研究内容。

此外，随着移动通信和数字服务的兴起，无论是 Internet、无线传感器网络，还是移动网络和流媒体服务，都对安全组通信协议的设计带来了颇多挑战，如开放设计的 Internet、易受入侵和非法攻击的移动网络、不可靠的无线传感器网络、流媒体系统的视频安全等因素。这使得组通信的应用将变得更加普遍，同时也对安全的组通信协议设计提出了许多新的并需亟待解决的课题。因此，有必要对组密钥管理进行研究，解决组通信中的安全问题。

作者在网络安全领域进行了一系列深入而系统的研究工作，本书主要对网络安全机制中的访问控制、身份认证、数字签名、密钥管理和视频安全技术进行全面、深入的阐述，书中绝大部分内容取材于我们近期在国际、国内一流学术期刊发表的论文，全面、系统地展示了很多新的研究成果和进展。

组织结构

本书主要对5种网络安全控制机制加以介绍，在结构上分为5章：

第1章是访问控制机制，首先概述了访问控制的发展过程，并对典型的访问控制模型进行了介绍，如自主访问控制、强制访问控制和基于角色的访问控制模型。然后基于Petri网对强制访问控制模型进行了安全性分析。针对移动网络的特点，提出了支持移动IPv6的访问控制模型，介绍方案的实现及其扩展。最后，对可信网络中的访问控制进行研究，根据可信网络中用户行为的可信模型，讨论了基于可信和信誉的访问控制机制。

第2章是认证机制，首先介绍RADIUS协议和AAA服务器的认证原理以及AAA在无线网络中的应用。然后针对大型、复杂的网络系统中存在着一系列相互信任或不相互信任的安全自治网络域，介绍多级安全域的认证模型，并用逻辑理论对安全域认证模型进行形式化描述。最后讨论了移动网络中的可以抵制DoS攻击的认证模型，通过性能和安全分析，证明该模型能够满足移动网络中安全性和可靠性的需求，并能抵制DoS攻击，极大地提高了认证协议的安全性。

第3章是数字签名机制，首先介绍公钥密码体制，对数字签名中的几个典型的机制进行阐述，如RSA，ElGamal，Schnorr和DSS数字签名机制。然后详细介绍椭圆曲线密钥体制，基于椭圆曲线提出了群体导向的数字签名方案，并对其进行安全分析和性能分析。

第4章是密钥管理机制，首先概述了组密钥分发机制研究现状，对集中式、分散式和分布式组通信密钥管理进行介绍。然后详细分析了基本的组密钥分发协议，并给出其安全性分析。提出了一个自愈的组密钥分发协议，并在此基础上讨论了基于时限用户撤销机制的自愈组密钥分发协议，给出了协议的具体应用和改进方案。最后，对无线传感器网络中的密钥管理进行了阐述和分析，介绍几个典型的密钥管理方案和协议，对其进行了综合分析，并给出了需要解决的研究问题。

第5章是基于应用层组播的视频安全机制，介绍流媒体与应用层组播、数字水印技术及视频加密技术，并提出了一个媒体相关的视频安全组播协议MSMP，详细介绍用户加入和退出机制，并对其可靠性和扩展性进行分析。最后讨论了视频流传输过程中的差错控制机制以及无线网络中多层非对等保护的动态优化组包策略。

本书特点与读者对象

本书具有以下鲜明特色。

(1) 完整性：内容丰富全面，结构合理，体系完整，将网络安全控制机制的5个方面，即访问控制、认证、数字签名、密钥管理和视频安全机制，进行全面和系统的介绍。

(2) 实用性：结合当前网络环境的特点，将网络安全控制机制应用于可信网络、移动网络和传感器网络，给出具体的应用实例，具有很强的实用性。

(3) 学术性：本书具有一定的理论高度和学术价值，书中绝大部分内容取材于作者近期在国际、国内一流学术期刊发表的论文，全面展示了大量网络安全方面最新的科研成果，具有很高的学术参考价值。

本书非常适合我国计算机网络和通信领域的教学、科研工作和工程应用参考。既可以供计算机、通信、电子、信息等相关专业的研究生和大学高年级学生作为教材或教学参考书，

也可以供计算机网络研究开发人员、网络运营商等网络工程技术人员参考。

致谢

作者的研究工作得到国家自然科学基金项目(Nos. 60673184，60503052，60429202，60432030)、国家重点基础研究发展计划("973"计划)项目(No. 2006CB708301)和国家高技术研究发展计划("863"计划)项目(Nos. 2006AA01Z218，2006AA01Z225)等的连续资助，在此表示深深的谢意!

西安第二炮兵工程学院的封富君博士(林闯的学生)在本书的写作过程中做了大量细致而辛苦的工作，在此对其表示衷心的感谢!

由于作者水平所限，加之计算机网络安全控制机制的研究仍处于不断的发展和变化之中，书中错误和不足之处在所难免，恳请专家、读者予以指正。

作 者
2008 年 5 月
北京　清华园

CONTENTS

目 录

Chapter 第 1 章

访问控制

访问控制技术起源于 20 世纪 70 年代，当时是为了满足管理大型主机系统上共享数据授权访问的需要。但随着计算机技术和应用的发展，特别是网络应用的发展，这一技术的思想和方法迅速应用于信息系统的各个领域。在 30 多年的发展过程中，先后出现了多种重要的访问控制技术，如自主访问控制(discretionary access control，DAC)、强制访问控制(mandatory access control，MAC)和基于角色的访问控制(role-based access control，RBAC)，它们的基本目标都是防止非法用户进入系统和合法用户对系统资源的非法使用。访问控制技术作为实现安全操作系统的核心技术，是系统安全的一个解决方案，是保证信息机密性和完整性的关键技术，对访问控制的研究已成为计算机科学的研究热点之一。

本章对不同网络环境下的访问控制进行研究，给出了针对不同网络的访问控制模型。首先概述了访问控制的基本目标、发展过程、分类及其研究趋势，然后基于着色 Petri 网对强制访问控制进行形式化描述和安全分析。针对移动通信网络，给出了支持层次移动 IPv6 的访问控制方案。最后，研究下一代网络发展的必然趋势，即可信网络下的访问控制及其实现机制。

1.1 访问控制概述

国际标准化组织(ISO)在网络安全标准 ISO 7498—2 中定义了 5 种层次型安全服务：身份认证服务、访问控制服务、数据保密服务、数据完整性服务和不可否认服务。其中访问控制是信息安全的一个重要组成部分，作为系统安全的关键技术，访问控制是一个老生常谈的内容同时又面临着新的挑战。随着网络技术的发展，访问控制技术也将作为网络安全的一个重要方面日益受到更多人的关注。授权和认证是访问控制的基础，正确的授权实际上依赖于认证。认证是决定一个用户的身份是否合法的过程。授权决定一个用户是否有权访问系统资源。一个信息系统必须维护一些用户 ID 和系统资源之间的关系，建立一个授权用户被允许访问的资源列表。访问控制技术不仅包括授权和认证，还可以有很多其他形式，如智能卡、密钥锁、生物信息识别(如指纹、视网膜或人脸)等。

1.1.1 访问控制基本概念

任何访问控制模型都会用到用户(user)、主体(subject)、客体(object)、操作(operation)和权限(permission)的概念,下面对这几个概念进行简单的介绍。

用户:被授权使用计算机的人员。一个用户可能有多个 ID,而这些 ID 可能被同时激活。一个用户的会话实例称为会话(session)。

主体:可以被其他实体施加动作的主动实体。主体可以是用户或其他任何代理用户行为的实体(如进程、作业和程序)。一个用户可以有多个主体,即使该用户只有一个会话。

客体:接受其他实体动作的被动实体。客体可以是一个可识别的资源,一个客体可以包含另一个客体。一个实体可以在某一时刻是主体,而在另一时刻是客体,这取决于该实体的功能是动作的执行者还是被执行者。

操作:由主体激发的主动进程。每个访问控制模型都与信息流相关,但是基于角色的访问控制要求主体和操作区别开来。

权限:在受系统保护的客体上执行某一操作的许可。在客体上能够执行的操作通常与系统的类型有关,权限是客体和操作的联合。两个不同客体上的相同操作代表着两个不同的权限,单个客体上的两个不同操作代表着两个不同的权限。

此外,最小特权(least privilege)原则是系统安全中最基本的原则之一。所谓最小特权原则是指:用户所拥有的权力不能超过他执行工作时所需的权限,即每个主体(用户和进程)完成某种操作时必不可少的特权。只给予主体"必不可少"的特权,一方面保证所有的主体都能在所赋予的特权之下完成所需要完成的任务和操作;另一方面,限制了每个主体所能进行的操作。最小特权原则在保持完整性方面起着重要的作用,实现最小权限原则,需分清用户的工作内容,确定执行该项工作的最小权限集,然后将用户限制在这些权限范围之内。在基于角色的访问控制中,只有角色需要执行的操作才授权给角色。当一个主体要访问某个资源时,如果该操作不在主体当前活跃角色的授权操作之内,则该访问将被拒绝。

坚持最小特权原则要求用户在不同的时间拥有不同的权限级别,这依赖于所执行的任务或功能。在某些环境和权限下,不必要的权限有可能会增加用户的额外负担,因此必须限制权限。然而过多的权限有可能会泄露信息,因此为了保证系统的机密性和完整性,必须避免赋予多余的权限。

1.1.2 访问控制目标

访问控制只是系统安全的一个解决方案,为了更好地理解访问控制的目标,有必要了解信息系统的风险。信息系统的安全风险可分为 3 类:机密性、完整性和有效性,记为 CIA。

机密性(confidentiality):保持信息的安全和私有,防止信息泄露给未授权的用户;

完整性(integrity):防止信息被非法用户篡改或破坏;

可用性(availability):保障授权用户对系统信息的可访问性。

访问控制是保证信息机密性和完整性的关键技术。机密性要求只有授权的用户可以读取信息。一般来说，系统中的某些信息是非常重要的，如军事上的某些数据，公司的财务信息及个人的账户信息等，这些信息都对机密性要求较高。完整性要求只有授权的用户可以在授权的方式下修改信息，是为了维护系统资源处于一个有效的、预期的状态，防止资源被不正确、不适当地修改，或维护系统不同部分的数据一致性。访问控制并不能完全保证可用性，它的作用是，当一个非法的攻击者试图访问系统时，有可能会受到阻止。

1.1.3 访问控制发展过程

从 1960 年起安全问题就引起了人们的关注，最早由 Lampson[1] 提出了访问控制的形式化机制描述，引入了主体、客体和访问矩阵的概念。对访问控制模型的研究，从早期的 20 世纪六七十年代至今，大致经历了以下 4 个阶段：

1. 20 世纪六七十年代应用于大型主机系统中的访问控制模型，较典型的是 Bell-Lapadula 模型[2]（简称 BLP 模型）和 HRU 模型[3]。

（1）Bell-Lapadula 模型

Bell-Lapadula 模型，简称 BLP 模型，由 Bell 和 Lapadula 将军队访问控制规则融入数学模型，定义推理计算机系统的安全性。该模型指出，进程是整个计算机系统的一个主体，它需要通过一定的安全等级来对客体发生作用。进程在一定条件下可以对诸如文件、数据库等客体进行操作。其安全规则指出，用户仅能访问安全级等于或低于用户安全级的那些信息。这是一个简单的策略，容易被人理解，但是在计算机系统上实现这个策略则是很困难的。无法预料的系统漏洞和系统中不同组件的交互，使得计算机系统具有安全脆弱性。在该模型中，计算机系统中的实体被分成抽象的对象。安全状态得到了详细的说明，而且通过从一个安全状态转到另一个安全状态的方式来证明状态转移过程仍然是安全的，进而归纳证明了该系统是安全的。

BLP 模型有两个基本的规则：简单安全规则和 *-特性，通常称为“不上读”和“不下写”。简单安全规则指出：实体不能读取安全级别高于它的对象，即实体的安全级别必须大于等于对象的安全级别。*-特性（星特性）指出：如果对象执行写操作，实体的安全级别必须小于等于对象的安全级别。

$$\text{read: } SL(\text{Entity}) \geqslant SL(\text{Obj}) \quad \text{简单安全规则}$$

$$\text{write: } SL(\text{Entity}) \leqslant SL(\text{Obj}) \quad *\text{-特性}$$

其中，SL 表示实体或对象的安全级别。

BLP 模型的核心思想是在系统中设置多个安全等级（如普通、秘密、机密和绝密），并要求系统中的所有存取操作必须遵守模型给出的保护信息安全的规则，以此实现强制存取控制，防止具有高安全级别的信息流入低安全级别的客体。BLP 模型不能直接用于商业系统，主要应用于军事系统。虽然 BLP 模型为通用的计算机系统定义了安全属性，且这种模型比较容易实现，但“不上读”和“不下写”的规则忽略了完整性，而使非法越权篡改成为可能。

BLP 模型已成为计算机安全基础的研究对象，该模型的发展影响了许多其他模型的发展，甚至很大程度上影响了计算机安全技术的发展，并渗透到计算机安全建模的所有策略，

它是第一个将实际系统的属性转化为规则的属性模型。在 BLP 模型的基础上，形成了很多标准，其中包括美国国防部的可信计算机评估标准。尽管该模型存在很多争议，但是它促进了计算机安全基础领域的进一步研究。

虽然 BLP 安全模型控制了对信息的写操作，保护了系统的机密性，但是多级安全策略并没有阻止对信息的非法修改。因此在 BLP 安全模型之后，用户很快认识到需要这样一种模型：能够阻止高安全级的进程读取低安全级的信息，而且进程不被低安全级的信息所影响。

Biba 完整性模型[4]是 1977 年提出的，是 BLP 模型的副本。BLP 模型着重系统的机密性，而 Biba 完整性模型则着重保证对象的完整性。Biba 模型将主体和客体按照强制访问控制系统进行分类，这种分类方法一般应用于军事用途。数据和用户被划分为 5 个安全等级：公开（unclassified）、受限（restricted）、秘密（confidential）、机密（secret）和绝密（top secret）。Biba 完整性模型确保实体只能向安全级别比它低的对象写信息，避免了在 BLP 模块中易发生的一种情况：安全级别高的实体可能故意破坏安全级别低的对象，并且实体可以从安全级别比它高的对象中读取信息。因为该模块只需要保证完整性，它是从完整性等级的方面被描述的，而不是从安全性或敏感性等级方面。这些规则可以总结为：

write：$IL(Entity) \geqslant IL(Obj)$　简单完整性规则

read：$IL(Entity) \leqslant IL(Obj)$　*-特性

Biba 模型基于两种规则来保障数据的完整性和保密性：

下读（no-read-up）：主体不能读取安全级别低于它的数据；

上写（no-write-down）：主体不能写入安全级别高于它的数据。

Biba 模型并没有被用来设计安全操作系统，因为 Biba 模块中的所有模块可以读任意级别比它高的对象，可以发送信息给级别比它低的对象，这可能造成实体泄露高级别对象的内容，在一定程度上忽视了保密性。但大多数完整性保障机制都是基于 Biba 模型的两个基本属性构建的。

（2）HRU 模型

1976 年 Harrison，Ruzzo 和 Ullman 提出 HRU 模型[3]，提供了更改访问权限的策略和创建以及删除主题和对象的权限，并指出用传统的访问矩阵并不能保证系统的安全性，即安全需要是安全的并不能说明系统的配置是安全的。用户可以放弃访问权限，也可以授权给其他用户，其他用户又可以授权给另外的用户，因此当权限一级级地被传递时，系统无法保证非授权的用户不会非法得到访问权限。

2. 美国国防部在 1985 年公布的可信计算机安全评价标准（TCSEC）[5]中明确提出了访问控制在计算机安全系统中的重要作用，并指出一般的访问控制机制有两种：自主访问控制（DAC）和强制访问控制（MAC）。目前 DAC 和 MAC 被应用在很多领域，关于 DAC 和 MAC 的相关内容将在 1.1.4 节中介绍。

3. 从 1992 年最早的 RBAC 模型，即 Ferraiolo-Kuhn 模型[6]的提出，到 Sandhu 等人对 RBAC 模型的研究，先后提出了 RBAC96[7]，ARBAC97[8]，ARBAC99[9]模型，一直到 2001 年的 NIST RBAC 标准[10]。

Ferraiolo-Kuhn 模型将现有的面向应用的方法应用到 RBAC 模型中，是基于角色的访

问控制(RBAC)最初的形式化描述,它对主体-角色活动(subject-role activation)、主体-客体(subject-object)关系、用户-角色(user-role)关系和角色集活动(role-set activation)进行了描述。有以下 3 个基本规则:

规则 1　角色分配(role assignment):当一个主体被分配了一个角色时,该主体才能执行一个事务。身份认证过程并不是一个事务,而系统中用户的其他行为都是通过事务完成的,因此,活动用户需要有一些活动角色。

规则 2　角色授权(role authorization):一个主体的活动角色必须授予该主体。由规则 1,这条规则保证了用户只能执行被授权的角色。

规则 3　事务授权(transaction authorization):当一个事务被授予一个主体的活动角色时,该主体才能执行该事务。由规则 1 和规则 2,该规则保证了用户只能执行被授予的事务。

RBAC 模型的正式描述见表 1.1.1,其特点是所有访问都是通过角色来实现的。一个角色实质上是权限的集合,所有用户通过分配的角色来接受权限。角色是相对稳定的,而用户和权限则可能变化很快,通过角色对访问进行控制简化了管理。RBAC 关系图如图 1.1.1 所示。

表 1.1.1　Ferraiolo 对 RBAC 的形式化描述

RBAC 的形式化描述
活动角色:AR(s: subject)={主体 s 的当前活动角色}
角色授权:RA(s: subject)={系统授给主体 s 的角色}
事务授权:TA(r: role)={系统授给角色 r 的事务}
exec(s: subject,t: tran)=true　当且仅当主体 s 有权执行事务 t,否则为 false 角色分配:$\forall s$: subject,t: tran • exec(s,t)$\Rightarrow$AR(s)$\neq\varnothing$ 角色授权: $\forall s$: subject • AR(s)$\subseteq$RA(s)

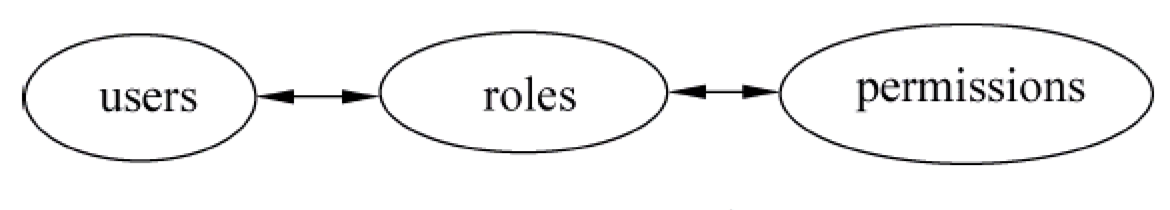


图 1.1.1　RBAC 关系图

多数计算机系统的访问控制是通过访问控制表(access control list, ACL)来实现的,所以系统资源,如文件、打印机和终端,都有一个授权用户列表,这样很容易回答"哪些用户可以访问客体 *X*",但是却很难回答"用户 *X* 能够访问哪些客体"。后者的回答需要扫描系统中数以百万计的客体并记录访问控制列表,而这个过程在实际的系统中可能会需要一天的时间。这个机制的特点是:ACL 可以很容易地给客体增加权限,但很难激发一个用户的所有权限。

在一些系统中,用户被分为组,称为实体(entry)。RBAC 和组的概念有些相似,组是用户的集合而不是权限的集合,权限是与用户和用户所属的组相关联的,如图 1.1.2 所示。由于用户通过 UID(user ID)或 GID(group ID)来访问客体,因此,当组权限从客体上撤销时,一旦权限被激活,用户可能重新获得访问权限。RBAC 要求通过角色进行访问加强了系统的安全性。

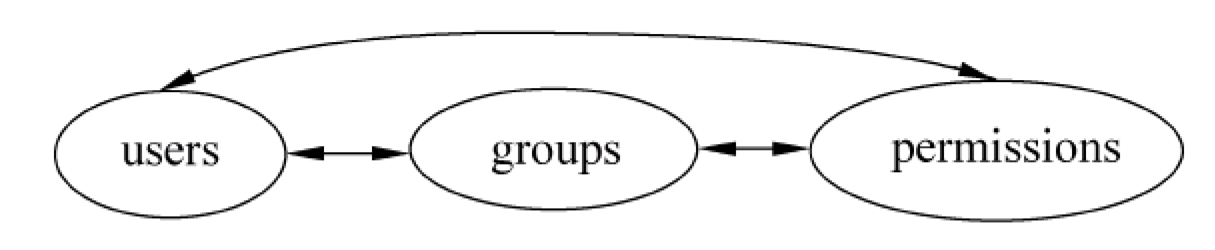


图 1.1.2 组访问控制关系图

Ferraiolo-Kuhn 模型的第 2 个重要特点是角色是分等级的：角色能够从其他角色中继承权限。此外，该模型包含了 Clark-Wilson 模型[11]。此后 NIST RBAC 参考模型对角色进行了详细的研究，在用户和访问权限之间引入了角色的概念，为 RBAC 模型提供了参考。关于 NIST RBAC 模型将在 1.1.4 节中加以介绍。

4. 此后，对访问控制模型的研究扩展到更多的领域，比较有代表性的有：应用于工作流系统或分布式系统中的基于任务的授权控制模型（TBAC）[12]，基于任务和角色的访问控制模型（T-RBAC）[13]，以及被称作下一代访问控制模型的使用控制（usage control，UCON）模型[14,15]，也称 ABC 模型[16]。UCON 模型不仅包含了 DAC，MAC 和 RBAC，而且还包含了数字版权管理（DRM）、信任管理等，涵盖了现代商务和信息系统需求中的安全和隐私这两个重要的问题，因此，UCON 模型为研究下一代访问控制提供了一种新方法，被称作下一代访问控制模型。

1.1.4 访问控制分类

访问控制策略是面向应用的，可以跨越多个计算平台，可以基于最小特权、权能、认证、责任或利益冲突。访问控制策略往往是动态变化的，是随着商业因素、政府规则和环境条件的变化而发生变化的，而策略需求在系统设计时是无法完全确定的，因此系统必须按照不断变化的策略加以设计。目前一般的访问控制策略有 3 种：自主访问控制（DAC）、强制访问控制（MAC）和基于角色的访问控制（RBAC）。

1.1.4.1 自主访问控制

自主访问控制（DAC）是一种最普遍的访问控制安全策略，最早出现在 20 世纪 70 年代初期的分时系统中，基本思想伴随着访问矩阵被提出，在 UNIX 类操作系统中被广泛使用。DAC 主要是为多用户的数据库系统设计的，系统用户改变较少，并且所有的资源都由一个实体来控制，通过用户身份或用户所属的组对客体的访问进行限制，具有主动访问资源的用户和主体有能力将信息传递给另一个主体。DAC 的核心思想是主体的拥有者通常是它的建立者，可以主动授权给其他人访问该主体，故 DAC 又称为基于主体的访问控制。

1. DAC 实现

DAC 是目前计算机系统中实现最多的访问控制机制。它的实现方法一般是建立系统访问控制矩阵，矩阵的行对应系统的主体，列对应系统的客体，元素表示主体对客体的访问权限。为了提高系统性能，在实际应用中常常是建立基于行（主体）或列（客体）的访问控制方法。基于行的方法是在每个主体上都附加一个该主体可以访问的客体的明细表，有 3 种实现形式：权能表、前缀表和口令。基于列的自主访问控制是对每个客体附加一个可访问它的主体的明细表，有两种实现形式：访问控制表（ACL）和保护位，其中使用最多的是访问控制表。

(1) 访问控制表(ACL)

ACL 可以决定任意一个主体是否能够访问该客体,它是通过在客体上附加一个主体明细表的方法来表示访问控制矩阵。表中的每一项包括主体的身份和对客体的访问权。ACL 是实现自主访问控制的最好的方法。访问控制系统通过检测 ACL 来决定访问是被授权或拒绝。表 1.1.2 是一个访问控制矩阵,表 1.1.3 是表 1.1.2 相应的 ACL。

表 1.1.2　访问控制矩阵

	客　　体		
主体	File_1	File_2	File_3
Chris	Read, Write		Write
Frank		Execute	
Bob	Read		Read

表 1.1.3　对应表 1.1.2 的 ACL

客体		
File_1	Chris: Read, Write	Bob: Read
File_2	Frank: Execute	
File_3	Chris: Write	Bob: Read

ACL 的一个优点是可以很容易地看到用户对客体的访问及操作,而且通过简单地删除 ACL 实体就可以取消对客体的访问,这些特点使 ACL 成为实现面向对象的 DAC 策略的理想方法;ACL 的另一个优点是,如果组中的用户有相同的访问权限,则在组后附加客体,代替组中的所有成员,这样使 ACL 不必很长。

(2) 权能表(capabilities list)

权能表决定用户是否可以对客体进行访问以及进行何种形式的访问(读、写、改、执行等)。一个拥有某种权力的主体可以按一定方式访问客体,并且在进程运行期间访问权限可以添加或删除。使用权能表实现的访问控制系统可以很方便地查询某一个主体的所有访问权限,只需要遍历这个主体的权能表即可,然而要查询对某一个客体具有访问权限的主体的信息是很困难的,必须查询系统中所有主体的权能表。此外,对权能表的检查很难撤销某主体对客体的访问,这使得权能表在商业上使用并不是很普遍。表 1.1.4 是对应表 1.1.2 的权能表。

表 1.1.4　对应表 1.1.2 的权能表

主体		
Chris	File_1: Read, Write	File_3: Write
Frank	File_2: Execute	
Bob	File_1: Read	File_3: Read

20 世纪 70 年代很多操作系统的访问控制安全机制是基于权能表实现的,但并没有取得商业上的成功。现代的操作系统大多改用基于 ACL 的实现技术,只有少数实验性的安全操作系统使用基于权能表的实现技术。在一些分布式系统中,也使用了权能表和 ACL 相结合的方法来实现访问控制机制。

(3) 前缀表(profiles)

前缀表包括受保护的客体名以及主体对它的访问权。当主体要访问某客体时,自主访问控制系统将检查主体的前缀是否具有它所请求的访问权。

(4) 保护位(protection bits)

保护位机制类似于 ACL,位与客体相关,而不是与用户或操作相关。保护位将用户分为 3 类:自身(self),文件的拥有者;组(group),对文件共享访问的用户的集合;其他(other),除拥有者和组成员之外的任何人。

访问控制系统中用户对文件的访问有 read(r),write(w)或 execute(x)操作。例如,假设一个文件的保护位是:(rwx) (r-x) (--x),则说明文件的拥有者对文件有读、写和执行的权限,组中成员有读和执行权限,其他人有执行权限。由于保护位机制不能完备地表达访问控制矩阵,因而很少使用。

2. DAC 的优缺点

DAC 根据用户的身份及允许访问权限决定其访问操作。在这种机制下,文件的拥有者可以指定系统中的其他用户或用户组对该文件的访问权。这种访问控制机制的灵活性较高,被广泛用于商业领域,尤其是在操作系统和关系数据库系统上。DAC 的优势是:①能够在一定程度上实现权限分离和资源保护;②使信息可以从被写的客体流向被读的客体;③用户可以自主地授予和撤销其他用户的访问权限。

然而也正是由于这种灵活性导致系统的信息安全性能降低。DAC 的缺点是:授权读是可传递的,一旦访问权被传递出去将难以控制,使访问权的管理相当困难,从而带来严重的安全问题;DAC 机制易遭到特洛伊木马攻击;在大型系统中,主、客体的数量巨大,采用 DAC 将使系统开销大到难以支付的程度。

1.1.4.2 强制访问控制

由于自主访问控制不能抵御特洛伊木马的攻击,强制访问控制(MAC)作为一种基于格(lattice-based)的访问控制策略应运而生。MAC 最早被应用在军方系统中,在军事和安全部门中应用较多。客体有一个包含等级的安全标签(如不保密、限制、秘密、机密、绝密),访问者拥有包含等级列表的许可,其中定义了可以访问哪个级别的客体,其访问策略是由授权中心决定的强制性的规则。MAC 的本质是基于格的非循环单向信息流政策,通过无法回避的存取限制来阻止直接或间接的非法入侵。它的两个关键规则是:不向上读和不向下写,即信息流只能从低安全级向高安全级流动,任何违反非循环信息流的行为都是被禁止的。

MAC 同样具有一些弱点:对用户恶意泄露信息无能为力;虽然 MAC 增强了信息的机密性,但不能实施完整性控制,而网络应用对信息完整性具有较高的要求,因此 MAC 可能无法胜任某些网络应用;在 MAC 系统中,实现单向信息流的前提是系统中不存在逆向潜信道,否则会导致信息违反规则的流动,这就给系统增加了安全性漏洞。此外,MAC 过于强调保密性,对系统的授权管理不够灵活。

Brewer-Nash 模型[17]是 Brewer 和 Nash 开发的用于商业领域的访问控制模型,它使用了一种简单且易于描述的 Chinese Wall 策略,最初是为投资银行设计的,但也可以应用于其他相似的场合。与 Bell-Lapadula 模型类似,Brewer-Nash 模型并没有明显区分用户和主体的概念,认为主体包括用户和以用户身份活动的进程,而且 Brewer-Nash 模型的写规则考虑到了特洛伊木马的可能性。两个模型的不同之处在于:在 Bell-Lapadula 模型中,读和写规则应用于主体-用户会话的整个生命周期中;而 Brewer-Nash 模型的读规则应用于用户的生命周期中,一旦用户读了数据集中的一个客体,该用户将不能读那些属于相同利益冲突集中的另一个数据集中的客体。

此外，美国 Secure Computing 公司提出了 TE(type enforcement)访问控制技术，该技术把主体和客体分别进行归类，它们之间是否有访问授权由 TE 授权表决定，TE 授权表由安全管理员负责管理和维护。授权关系表(authorization relations)是对应于访问矩阵中每个非空元素的实现技术，它的每一行就是访问矩阵中的一个非空元素，是某个主体对应于某个客体的访问权限信息。如果授权关系表按主体排序，查询时就可以得到权能表的效率；如果按照客体排序，查询时就可以得到 ACL 的效率。

1.1.4.3　基于角色的访问控制

随着网络的发展和 Internet 的广泛应用，信息的完整性需求超过了机密性，传统的 DAC/MAC 策略已无法满足信息完整性的要求，于是提出了基于角色的访问控制(RBAC)。RBAC 发展到现在已较为成熟，并且在许多大型系统中得以实现。目前 RBAC 是一个研究的热点，其中以美国 George Mason 大学的 Sandhu 等人提出的基于角色的访问控制模型 RBAC96[7]，ARBAC97[8] 和 ARBAC99[9] 影响较大。2001 年 8 月 NIST 发表了 RBAC 建议标准[10]，该建议标准综合了该领域众多研究者的研究成果，描述了 RBAC 系统最基本的特征，旨在提供一个权威的、可用的 RBAC 参考规范，为 RBAC 的进一步研究指明了方向。

1. NIST RBAC 标准

NIST 包括两部分内容：RBAC 参考模型和 RBAC 功能规范。

RBAC 参考模型给出了 RBAC 集合和关系的严格定义，包括 4 部分内容：核心 RBAC (core RBAC)、等级 RBAC(hierarchical RBAC)、静态职责分离(static separation of duties, SSD)和动态职责分离(dynamic separation of duties, DSD)。这 4 个模型为扩展 RBAC 提供了一个基本的参考。每个模型的定义包括：一个基本元素集、元素集合间的 RBAC 关系和一个映射函数集。

RBAC 功能规范为每个组件定义了关于创建和维护 RBAC 集合和关系的管理功能、系统支持功能和审查功能。其中管理功能用于创建和维护构成 RBAC 模型构件的各种系统要素及相互关系；系统支持功能用于在用户与系统交互时支持 RBAC 模型构建的各种功能，如建立会话、添加/删除活跃角色、确定访问逻辑等；审查功能用于审查由管理功能和系统支持功能产生的各种活动的结果。

核心 RBAC 的基本思想是通过角色建立用户和访问权限的多对多关系，定义了能够构成一个 RBAC 访问控制系统的最小的元素集合，由 5 个基本元素组成：用户(users)、角色(roles)、权限(PRMS)、对象(objects)和操作(OPS)。图 1.1.3 为核心 RBAC 模型，其中角色分配(user assignment，UA)和权限分配(permission assignment，PA)是多对多的关系，用双箭头表示。一个用户可以是多个角色的一个成员，一个角色可以拥有多个用户。同样，一个角色可以有多个权限，相同的权限可以分配给多个角色。

为了便于对 RBAC 系统中复杂的权限进行管理，等级 RBAC 引入角色间的继承关系(role hierarchy，RH)，即一个角色可以通过继承其他一个或多个角色来定义。角色等级是一个严格意义上的偏序关系，根据偏序关系中有无限制又可分为一般继承 RBAC(general hierarchical RBAC)和受限继承 RBAC(limited hierarchical RBAC)两种。一般继承支持任

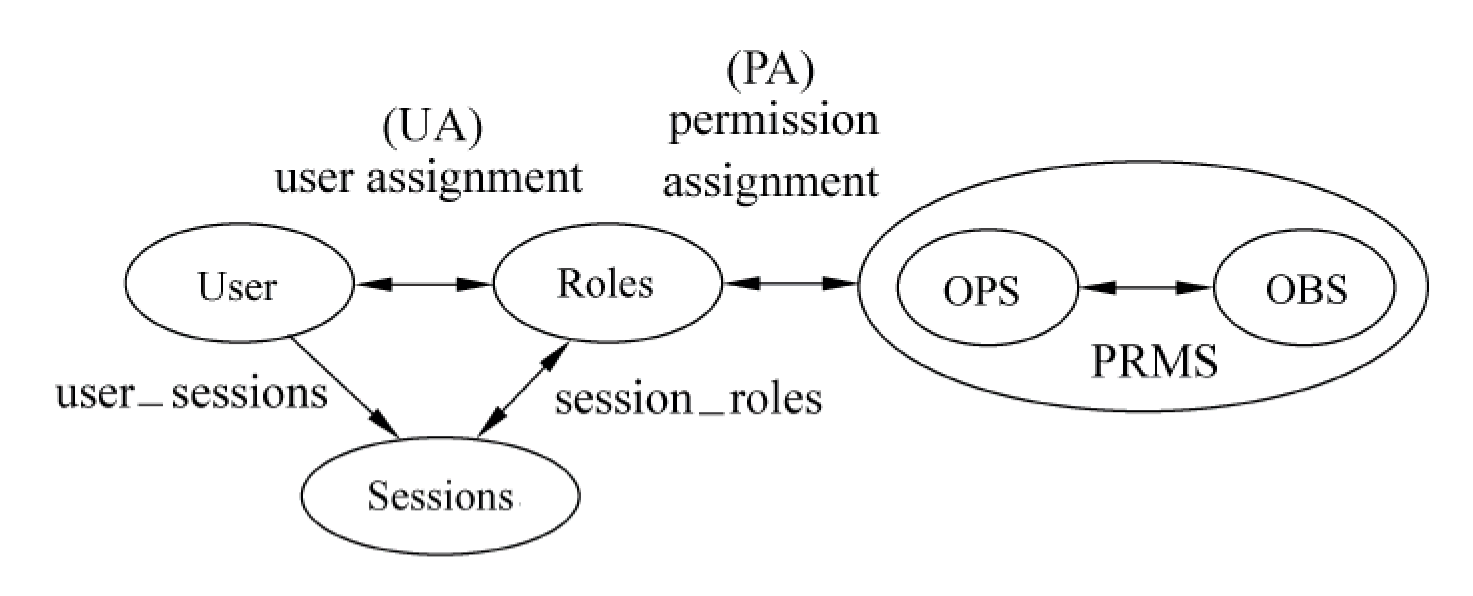


图 1.1.3 核心 RBAC 模型

意的具有偏序关系的角色层次结构,包括支持权限的多继承关系和用户成员的多继承关系等。在受限继承关系中,角色之间不存在多继承。

约束 RBAC 引入了职责分离(separation of duties,SoD)的概念,其目的是为了防止用户的操作超出其许可范围而导致欺骗和错误的发生。SoD 分为两种:静态 SoD(static separation of duties, SSD)和动态 SoD(dynamic separation of duties, DSD)。SSD 是用于解决角色系统中潜在的利益冲突(conflict-of-interest)的策略,是在为用户委派角色时,角色集与角色继承之间的一种约束机制。

DSD 也用于限制用户的访问权限,但与 SSD 作用机制不同,DSD 是对用户会话中可激活的角色进行约束,是对最小权限原则的扩展,每个用户根据其执行的任务可以在不同的环境下拥有不同级别的访问权限。DSD 中用户可以被授予多个角色,包括有冲突的角色,但它们不能在同一个会话中被激活。DSD 约束可视作一个二元组(role_set,n),表示任何用户在某个角色子集中不能同时激活 n 个以上的角色。

2. RBAC 的特点

RBAC 最突出的优点就在于系统管理员能够按照部门、企业的安全政策划分不同的角色,执行特定的任务。一个 RBAC 系统建立起来后,主要的管理工作即为授权或取消用户的角色。用户的职责变化时,只需要改变角色即可改变其权限;当组织的功能变化或演进时,则只需删除角色的旧功能,增加新功能,或定义新角色,而不必更新每一个用户的权限设置。这极大地简化了授权管理,使对信息资源的访问控制能够更好地适应特定单位的安全策略。RBAC 已被广泛应用于数据库系统和分布式资源互访中。

RBAC 的另一个优势体现在为系统管理员提供了一种比较抽象的、与企业通常的业务管理类似的访问控制层次。通过定义(建立)不同的角色、角色的继承关系、角色之间的联系以及相应的限制,管理员可以动态或静态地规范用户的行为。

RBAC 的一个重要特性是策略中立,它是一种表达策略的方法而不是一个具体化的特定安全策略。RBAC 除支持最小特权原则、职责分离原则和数据抽象原则等众所周知的安全原则以外,还支持角色继承和角色互斥。

为了提高效率,避免相同权限的重复设置,RBAC 采用了"角色继承"的概念,定义了这样的一些角色:它们有自己的属性,但可能还继承其他角色的属性和权限。角色继承把角色组织起来,反映组织内部人员之间的职权和责任关系,角色可以拥有自己的属性和权限,也可以继承其他角色的属性和权限。

角色间的层次关系≥是一个偏序关系。存在两种继承关系:访问权限的继承关系和用

户的继承关系。权限的继承是自下而上的，上层角色可以继承下层角色的部分或全部权限，不仅可以访问本角色的资源，还可以访问下层角色的资源。用户的继承是自上而下的，上层角色的用户也是下层角色的用户，即下层角色的用户包含上层角色的用户。两种继承关系可以总结为：

（1）权限继承关系：如果 $r_1 \geqslant r_2$，那么 r_1 自动拥有 r_2 的所有权限。

（2）用户继承关系：如果 $r_1 \geqslant r_2$，那么即使没有把 r_2 分配给用户 U，拥有角色 r_1 的用户 U 也会自动拥有角色 r_2。

3. RBAC 扩展模型

角色概念的引入为访问控制策略带来了很多优点。但是在很多实际应用中，仅靠角色和权限的访问控制无法满足人们的需求，因此出现了很多 RBAC 的扩展模型。对基本 RBAC 的扩展有广义 RBAC（generalized RBAC，GRBAC）模型[18]、基于组（team-based）的访问控制模型[19~21]、基于内容（content-based）的访问控制模型[22]、基于代理的访问控制模型[23]等，这些模型都是针对特定的应用领域而设计的。

大多数 RBAC 模型都支持用户和角色的约束机制，且对 SoD 研究较多，对约束描述语言的研究很早就提了出来。Bertino 等人提出了一种基于逻辑的约束描述语言，可以用来描述角色、用户和工作流任务的约束[24]。Ahn 等人提出了一种基于角色的认证约束描述语言 RCL2000（Role-Based Constraints Language 2000）[25]，取代了 RCL2000 的早期版本 RSL99[26]。RCL2000 不仅能够描述 SoD 约束和禁止（prohibition）约束，而且能够描述义务（obligation）约束。RCL2000 具有强大的描述能力，任何用 RCL2000 描述的性质都可以转化为一阶谓词逻辑。这些约束描述语言都可以描述 SoD 约束的各种可能的情况，但都没有考虑到时态和状态，无法描述时态约束。

尽管 RBAC 自身有很多优点，但是在很多实际应用中，RBAC 并不能满足安全需求。如在实际组织中，进程可能只在某个时间段内有效，角色也在该时间间隔内被激活，这就要求用时态来描述角色活动。Bertino 等人提出了一个基于时间（time-based）的访问控制模型，支持非 RBAC 环境中的时态认证[27]。之后，Bertino 等人又提出了时态 RBAC（temporal RBAC，TRBAC）模型[28]，支持周期性的角色授权与撤销。TRBAC 是首先考虑到时态约束的模型，可以应用于与时态有关的数据库系统中。Atluri 等人提出了一个时态数据认证模型（temporal data authorization model，TDAM）[29]，能够基于数据的时态特点描述访问控制策略，但是 TDAM 并不支持角色约束。

虽然 TRBAC 将时态约束引入 RBAC 模型中，但是也存在一些缺陷：TRBAC 认为角色在不同的时间段内被授权或撤销，但没有考虑用户-角色和角色-权限的时态约束；TRBAC 没有很好地区分角色授权（enabling）和角色激活（activation）的概念，因此只能处理角色授权的约束，而无法处理角色激活的约束。此外，TRBAC 没有提到角色层次和 SoD 约束的时间语义。为了解决以上这些问题，Joshi 等人提出了广义时态 RBAC 模型（generalized temporal role-based access control，GTRBAC）[30,31]。GTRBAC 能够描述更加广泛的时态约束，如角色、用户-角色分配和角色-权限分配的周期描述。此外，GTRBAC 还提出了角色继承和 SoD 约束，而对继承和 SoD 约束的时态语义将成为今后的一个研究方向。

1.1.5 访问控制研究趋势

随着 Internet 技术、无线通信技术、电子技术及计算技术的高速发展，计算机系统的应用日益广泛。然而如此庞大的系统其脆弱性是不可避免的，计算机网络正面临着严峻的安全挑战，而访问控制作为保障网络安全的一个重要技术受到人们的密切关注。针对不同的网络环境，研究人员提出了不同的访问控制模型，这些模型都为网络安全的实现提供了很好的解决途径。可以看到访问控制技术的研究呈现出以下发展趋势：

(1) 分布式系统中的访问控制技术将成为未来的研究热点，包括适用于分布式系统或工作流系统的动态访问控制及分层访问控制，基于角色的访问控制将会在大型分布式系统中得到更加广泛的应用；

(2) 针对网络信息系统、无线网络(如 Ad hoc、传感器网络和移动网络)及 P2P 系统，需要灵活的、易扩展的、支持多种安全策略的访问控制技术，这将成为重要的研究方向，而基于上下文、基于语义、基于位置等的访问控制模型或者它们的相互结合，将会应用到更多的网络环境中；

(3) 可信网络是计算机网络发展的一个必然趋势，对可信模型、可信评估以及基于可信的访问控制的研究将成为重要的研究方向。此外，可信网络的安全问题已经远远不止保密性和完整性的问题，单一安全技术很难保证系统的真正安全。访问控制技术与其他安全技术进一步的结合，如访问控制与策略、域之间的隔离、密钥管理以及系统行为认证等技术的结合，将成为研究热点。

1.2 基于着色 Petri 网的强制访问控制模型

在计算机系统的安全研究中，安全模型是一种重要的形式化描述和验证方法，它不仅应用于系统的安全定义上，而且也应用于系统安全的设计和实现上。安全模型的重要性主要体现在：首先，它抽象而准确地描述了系统的安全需求而不涉及其实现细节，这使得我们能够全面而准确地理解系统的安全需求定义，并通过形式化的分析方法找到系统在安全上的漏洞，即系统脆弱性[32]；其次，安全模型是系统安全开发过程中的关键步骤，在美国国防部的“可信计算机系统的评价标准(TCSEC)”中，从 B 级开始就要求对安全模型进行形式化描述和验证，并作隐通道分析等；最后，安全模型的形式化描述和验证能够增强系统安全的可信度。

Petri 网是一种重要的数学工具，它能够有效地对信息系统进行形式化描述和建模。作为数学工具，Petri 网可以通过建立系统的状态可达图来分析系统的行为。Petri 网的模型分析方法具有坚实、严密的数学基础，因此，利用 Petri 网可以方便地对信息安全模型的特性进行形式化分析验证。文献[33]利用 Petri 网对信息流安全模型进行了分析；在文献[34]中，Marc 利用 Petri 网对基于 Take-Grant 的自主访问控制模型进行建模分析，并给出了从 Take-Grant 模型到 Petri 网模型的等价转换形式。另外，Petri 网在工作流模型中也得到了广泛应用。例如，文献[35]利用 Petri 网提出了一种适用于工作流环境中的上下文关

联的访问控制模型；文献[36]利用 Petri 网分析了工作流模型中的同步认证机制；而在文献[37～40]中，Knorr 则利用 Petri 网对工作流系统的动态访问控制进行了建模分析。

本节在安全级格模型和 BLP 安全模型的基础上，对强制访问控制的安全模型进行了形式化描述和定义，并给出了与其等价的着色 Petri 网模型。在 Petri 网可达图的基础上，深入探讨了强制访问控制模型的 4 种安全性质：主体对客体访问的时序特性、主体访问的可达性、因主体采用动态安全级访问而带来的安全隐患，以及因主体对客体的间接访问而导致客体关系的可推测性。最后，通过对一个安全模型的范例分析研究表明：基于 Petri 网的安全模型分析方法是一种重要的形式化分析工具，它可以充分利用现有 Petri 网的形式化分析方法和模型检测方法对系统安全模型的性质进行自动验证，并能够在安全模型的设计和实现阶段有效地改善系统的总体安全策略。

1.2.1 强制访问控制模型的形式化描述与安全分析

从安全的观点来看，一个计算机系统中存在有两类基本的实体：一类是客体，它主要包括文件、存储区这些不活跃的被动实体；另一类是主体，它主要包括用户、进程这些活跃的主动实体。客体是信息资源的载体和通道，主体通过发起对客体的请求访问而获取客体中的信息资源。强制访问控制的主要特点是系统中所有的主体和客体都有一定的安全级别，主体对客体的访问受到安全级别的影响。主体和客体的安全级由系统安全管理员决定，用户自己无权决定主体对客体的访问权限。

多级安全(multi-level security，MLS)是一个与 MAC 密切相关的概念。在 MAC 模型中，主体和客体分别与某个特定的安全级关联，系统所有的安全级构成了一个多级安全的体系结构。

1.2.1.1 多级安全的格模型

强制访问控制主要是通过安全级来实施，安全级包含“密级”和“部门集”两方面。其中，以人作为安全主体的部门集，表示他可以涉猎的信息范围；而以信息资源为主体的部门集则表示该信息所涉及的范围。

设 $D_0=\{d_i \mid i=1,2,\cdots,n\}$ 是系统全体“部门集”所构成的有限集，power(D_0)表示集合 D_0 的幂集，则全体“部门集”所构成的集合为 $D=\{D_i \mid D_i\in \text{power}(D_0)\}$。

定义 1.2.1 安全域 D 的格模型为 $\langle D,\oplus,\otimes,\varnothing,D_0\rangle$。其中空集 $\varnothing$ 和全体安全域 D_0 分别是代数系统的零元和单位元；$\forall D_i, D_j\in D$，则运算 $\oplus$，$\otimes$ 和偏序关系“$\leqslant$”的定义如下：

$$D_i \oplus D_j = D_i \cup D_j;\quad D_i \otimes D_j = D_i \cap D_j;\quad D_i \leqslant D_j \to D_i \subseteq D_j$$

显然 D 是一个格，序关系“$\leqslant$”是偏序关系，序对 $\langle D,\leqslant\rangle$ 是偏序集，满足自反性、传递性和反对称性。 □

定义 1.2.2 密级 S 的格模型定义为 $\langle S,\oplus,\otimes,\text{low},\text{high}\rangle$。其中，$S=\{S_i \mid i=1,2,\cdots,n\}$ 是密级的全序集；零元 low＝ S_1 是最低密级；单位元 high＝S_n 是最高密级；运算 $\oplus$，$\otimes$ 和全序关系“$<$”可以定义为

$$S_i \oplus S_j = S_{\max(i,j)};\quad S_i \otimes S_j = S_{\min(i,j)};\quad S_i < S_j \to i < j$$

因此，基于密级 S 和安全域 D 的格定义可以给出多级安全 L 的格模型定义。 □

定义 1.2.3 多级安全 L 的格模型为 $\langle L,\oplus,\otimes,(\varnothing,\text{low}),(D_0,\text{high})\rangle$。其中，$L$ 是 S 和 D 的直积，即 $L=S\times D=\{(S_i, D_i)\mid S_i\in S, D_i\in D\}$。$\forall (S_i, D_i),(S_j, D_j)\in L$，运算 $\oplus$，$\otimes$ 和偏序关系“$\leqslant$”的定义如下：

$$(S_i, D_i)\oplus(S_j, D_j)=(S_i\oplus S_j, D_i\oplus D_j)=(S_{\max(i,j)}, D_i\cup D_j)$$

$$(S_i, D_i)\otimes(S_j, D_j)=(S_i\otimes S_j, D_i\otimes D_j)=(S_{\min(i,j)}, D_i\cap D_j)$$

$$((S_i, D_i)\leqslant(S_j, D_j))\rightarrow((S_i<S_j)\wedge(D_i\leqslant D_j))$$ □

定理 1.2.1 多级安全 L 是一个布尔格，其中，零元和单位元分别是 $(\varnothing, \text{low})$ 和 (D_0, high)。 □

事实上，根据运算 $\oplus$，$\otimes$ 的定义有：$\forall L_i, L_j\in L$，$\text{Sup}(L_i, L_j)=L_i\oplus L_j\in L$；$\text{Inf}(L_i, L_j)=L_i\otimes L_j\in L$。另外，由偏序关系“$\leqslant$”的基本定义可知定理 1.2.1 成立。

多级安全 L 的元素 (S_i, D_i) 是一个二维向量，两个分量分别表示“密级”和“安全域”。安全级 L 是由系统的安全管理员定义。在如下的强制访问控制安全模型中，可定义系统密级为 $\{U, C, S, TS\}$，且满足全序关系 $U<C<S<TS$，它们分别表示 Unclassified(U)，Confident(C)，Secret(S)，TopSecret(TS)。相应地，安全级 L 可表示为 (U, D_i)，(C, D_i)，(S, D_i)，(TS, D_i)，其中 $D_i\in D$。为表示方便，在如下的模型分析中，我们分别用 U，C，S，TS 来表示相应的安全级 L，这并不影响对问题的结论分析。

1.2.1.2 强制访问控制模型(MAC)

MAC 模型具有较强的灵活性，由系统安全管理员给主体和客体分配不同的安全级别。在实施访问时，系统需要对主体和客体的安全级别进行比较，再决定主体访问客体的安全策略。主体和客体的安全级是一个二维向量：一个是具有偏序关系的密级，另一个是部门集。因此，基于多级安全 L 的格模型[41]和 Bell-LaPadula 模型的特点，我们可以形式化地描述 MAC 模型如下：

定义 1.2.4 MAC 模型是一个六元组 $M=(S, O, A, L, f, R)$，其中：

(1) S：主体集；O：客体集；

(2) A：$S\times O\rightarrow\{\varnothing,\{\text{read}\},\{\text{write}\},\{\text{read}, \text{write}\}\}$ 表示主体对客体的访问模式；

(3) L：安全级 $\langle L,\leqslant\rangle$ 是格，其元素 (S, D) 是一个二维向量，两个分量分别表示“密级”和“部门集”，并且满足 $((S_i, D_i)\leqslant(S_j, D_j))\rightarrow((S_i<S_j)\wedge(D_i\leqslant D_j))$；

(4) f：$S\cup O\rightarrow L$ 是主体或客体到安全级的映射。每个主体的安全级表示为 $(L_{cur}, L_{min}, L_{max})$，其中 L_{min}，$L_{max}\in L$ 分别表示主体 S 所允许的最大和最小安全级，而 $L_{cur}\in L$ 则表示系统安全管理员赋予主体 S 的当前安全级，且有 $L_{min}\leqslant L_{cur}\leqslant L_{max}$；$L_{obj}\in L$ 表示客体 O 的安全级；

(5) R：一组安全规则，规定主体对客体访问时应该遵循的约束条件。 □

模型的安全规则 R 是基于 BLP 模型的安全策略，即限制“向上读”和“向下写”，它们分别对应于 BLP 模型的简单安全规则和星规则。

① 简单安全规则：主体可以读客体，当且仅当 $L_{obj}\leqslant L_{max}$；

② 星规则：主体对客体只有“写(write)”权，当且仅当 $L_{cur}\leqslant L_{obj}$；主体对客体有“读(read)”权，当且仅当 $L_{obj}\leqslant L_{cur}$；主体对客体有“读写(read-write)”权，当且仅当 $L_{obj}=L_{cur}$。

实际上，简单安全规则是隐含在星规则中的，这是因为主体的当前安全级不大于主体的

最大安全级别。星规则主要针对的是不可信主体。由安全级 L 的格模型可知，系统中的任何两个元素（主体或客体）都可以比较大小，因此，可以根据主体和客体安全级别的高低对信息的流向加以控制。

定义 1.2.5 在 MAC 模型 M 中，系统管理员授予主体的安全级是一个范围（L_{min}，L_{max}）。若 $L_{min} < L_{max}$，则称该主体是可信主体；若 $L_{min} = L_{max}$，则称该主体是不可信主体。□

1.2.2 着色 Petri 网

Petri 网以研究系统的组织结构和动态行为为目标，着眼于系统中可能发生的各种变化之间的关系，它只关心变化所需条件和变化对系统状态的影响。正是由于 Petri 网所具备的强有力的表达能力、良好的数学基础，使得 Petri 网的应用范围日益扩大，且成为用来描述安全模型的较好工具。着色 Petri 网 CPN（colored Petri net）[42]是基本 Petri 网的一种扩展形式，它增强了 P/T 网对模型的模拟和描述能力。

定义 1.2.6 六元组 $\Sigma=(S, T, F, K, W, M_0)$ 是 Petri 网系统，当且仅当：

（1）S 是一个有限位置集，T 是一个有限变迁集，且 $S \cup T \neq \varnothing$，$S \cap T = \varnothing$；

（2）$F \subseteq (S \times T) \cup (T \times S)$；

（3）$K: S \to N^+ \cup \{\infty\}$ 是位置容量函数；

（4）$W: F \to N^+$ 是弧权函数；

（5）$M_0: S \to N$ 是初始标识，且满足 $\forall s \in S, M_0 \leqslant K(s)$。□

Petri 网系统的前置关联矩阵 Pre 和后置关联矩阵 $Post$ 为 $|P| \times |T|$ 矩阵：

（1）$Pre[s,t]=W(s,t)$，当且仅当：$s \in S, t \in T, (s,t) \subseteq F, w(s,t) > 0$；

（2）$Post[s,t]=W(t,s)$，当且仅当：$s \in S, t \in T, (t,s) \subseteq F, w(t,s) > 0$。

t 在 M 有发生权的条件是：$\forall s \in S$，有 $s \in {}^{\cdot}t \Rightarrow M(s) \geqslant W(s,t) \wedge s \in t^{\cdot} \Rightarrow M(s)+W(t,s) \leqslant K(s)$，这是说 M 授权 t 发生，记作 $M[t>$。若 t 在 M 有发生权，那么 t 就可以实施。发生的结果是把 M 变成新标识 M'，记作 $M[t>M'$ 或 $M \xrightarrow{t} M'$，M' 叫作 M 的后继标识。

定义 1.2.7 CPN 网定义为一个九元组 $CPN = (\Sigma, P, T, A, N, C, G, E, I)$，

（1）Σ 是一个有限非空的原子颜色集；

（2）P 和 T 分别是一个有限位置集和变迁集；

（3）A 是一个有限弧集：$P \cap T = P \cap A = T \cap A = \varnothing$；

（4）N 是一个节点函数：$A \to (P \times T) \cup (T \times P)$；

（5）C 是一个颜色函数：$p \to \Sigma$；

（6）G 是一个保证函数：$\forall t \in T$，$[\mathrm{Type}(G(t))=\mathrm{Boolean} \wedge \mathrm{Type}(\mathrm{Var}(G(t))) \subseteq \Sigma]$，与变迁 t 相关联的保证函数描述了该变迁能够发生的前提条件；

（7）E 是一个弧函数：$\forall a \in A$，$[\mathrm{Type}(E(a))=C(p(a))_{MS} \wedge \mathrm{Type}(\mathrm{Var}(E(a))) \subseteq \Sigma]$，其中 $p(a)$ 表示 $N(a)$ 的位置；

（8）I 是一个初始化函数。□

定义 1.2.8 CPN 中变迁 t 的绑定是一个定义在 $\mathrm{Var}(t)$ 上的函数 b，即：$\forall v \in \mathrm{Var}(t)$，$b(v) \in \mathrm{Type}(v)$。令 $B(t)$ 表示变迁 t 的所有绑定，$G(t)\langle b \rangle$ 表示保证函数 $G(t)$ 在绑定 b 上

的值。

定义 1.2.9 令 CPN=(Σ, P, T, A, N, C, G, E, I),则有:

(1) 一个变迁步 Y 在标识 M 下是可实施的,当且仅当:$\forall p\in P$, $\Sigma_{(t,b)\in Y}E(p,t)\langle b\rangle\leqslant M(p)$。若变迁步 Y 在标识 M 下可实施,且$(t,b)\in Y$,则称变迁 t 在标识 M 下,对绑定 b 而言是可实施的。

(2) 如果变迁步 Y 在标识 M 下是可实施的,那么 Y 可以实施并产生一个新的后继标识 M',M'可定义为:

$\forall p\in P,\quad M'(p)=(M(p)-\Sigma(t,b)\in Y\,E(p,t)\langle b\rangle)+\Sigma(t,b)\in Y\,E(t,p)\langle b\rangle$。

(3) 标识 M 经过 Y 的实施得到新的标识 M',可表示为 $M[Y>M'$。 □

定义 1.2.10 标识 M 是由 M_0 可达的,当且仅当存在一个变迁发生序列 σ,使得 M_0 经 σ 实施得到 M,亦即,$M_0[\sigma>M$。由 M 可达的标识集可表示为$[M>$。 □

定义 1.2.11 CPN 的可达图(发生图)是一个以标识为节点的有向图 $G=(V, A, N)$,其中:节点集 $V=[M_0>$;弧集 $A=\{(M_1, b, M_2)\mid M_1\in V\wedge M_2\in V\wedge b\in BE\wedge(M_1[Y>M_2])\}$;$N$ 是节点函数:$\forall a=(M_1, b, M_2)$, $N(a)=(M_1, M_2)$。 □

1.2.3 基于着色 Petri 网的强制访问控制模型

在基于 Petri 网的安全性分析中,状态空间的可达性分析是一种重要的安全属性验证方法。为了方便将原有的 MAC 模型转换成与其在语义上等价的 CPN 模型,首先需要引入实体安全模型 ESM(entity security model)的概念。与数据库中 EM(entity model)的概念相类似,ESM 模型更接近于 MAC 的现实模型,而 MAC 模型的 CPN 模型则是抽象层次更高的模型。因此,在 MAC 模型的转换框架模型中,如图 1.2.1 所示,ESM 模型能够在保证语义一致性的前提下将 MAC 模型映射成与之等价的 CPN 模型。



图 1.2.1 强制访问控制模型的转换框架模型

1.2.3.1 实体安全模型 ESM

定义 1.2.12 实体安全模型 ESM 是一个四元组 ESM=(Entities, Relations, f, L),其中 Entities 表示模型的实体集合;Relations 表示实体间的关系,且有 Entities$\in O$, Relations$\in O$(O 表示 MAC 模型中的客体集);L 表示安全级,且安全级$\langle L,\leqslant\rangle$是格;$f$: Entities$\cup$Relations$\rightarrow L$ 是客体到安全级的映射,即系统安全管理员赋予实体 Entities 和实体间的关系 Relations 的安全级。 □

考虑图 1.2.2 所示的实体安全模型,则有:实体集 Entities=(institute, Prof. 1, Prof. 2, symposium, project);关系集 Relations=(member, research, attend, subject, director);$L=(U, C, S, TS)$,且满足全序关系 $U<C<S<TS$;客体到安全级的映射关系 f,如 f(institute)$=U$;f(Prof. 1)$=U$ 等。

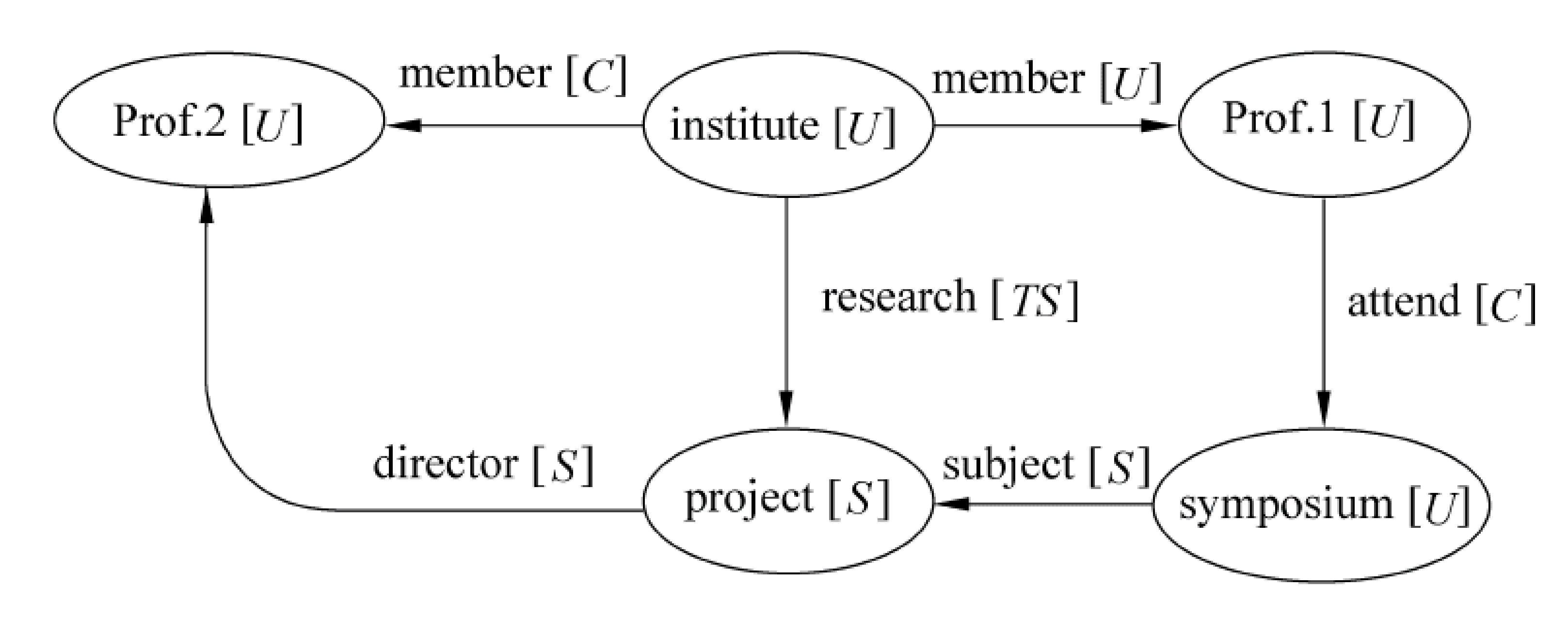


图 1.2.2　强制访问控制模型的等价实体安全模型

1.2.3.2　MAC 模型的等价 CPN 模型

根据 CPN 网的语义，为了生成与 ESM 模型等价的 CPN 模型，我们可以定义 CPN 模型和与之对应的 ESM 模型的映射转换算法如下：

（1）位置 P：对应于 ESM 模型中的 Entities。

（2）变迁 T：对应于 ESM 模型中实体之间的联系 Relations。

（3）原子颜色集：$\Sigma=\{L, \text{Access_mode}, PR\}$，其中 L 表示安全级；Access_mode 表示主体对客体的访问模式。变量 L_{max} 和 L_{cur} 分别表示主体的最大和当前安全级，L_{obj} 表示客体的安全级。

（4）颜色函数：$C(p)=PR=L\times L\times \text{Access_mode}$。

（5）变迁保证函数 Guard：与变迁相关的 Guard 函数描述了发生状态变迁的点火条件，也即主体对实体之间关系 Relations 拥有读权限的安全级约束条件。利用安全模型的星规则：主体可以读客体，当且仅当 $L_{obj}\leqslant L_{max}$。因此，我们可用式 $[L_{max}\geqslant L_{relation}]$ 来表示 Guard 函数，其中 $L_{relation}$ 表示实体间联系 Relations 的安全级别，且由系统安全管理员加以定义。

（6）弧函数：出弧 $A\in P\times T$ 表示从位置 $p\in P$（表示客体）移出相应的 token，即 $(L_{max}, L_{cur}, \text{access})$，出弧函数 $\text{out}(L_{obj})$ 决定 token 的数目和颜色值。入弧 $A\in T\times P$ 则表示在与该弧相连的位置 $p\in P$（表示客体）增加 token，入弧函数 $\text{in}(L_{obj})$ 决定 token 的数目和颜色值；入弧函数 $\text{in}(L_{obj})$ 实施主体对客体的访问实施强制访问控制（限制"向上读"和"向下写"），根据主体和与弧终端节点所表示客体的安全级别来确定主体对客体的访问模式 access。

定理 1.2.2　根据如上 ESM 模型和 CPN 网之间映射算法所生成的 MAC 模型的 CPN 网，它在语义上与 ESM 模型所描述的 MAC 模型等价。□

因此，根据如上 CPN 模型和与之对应的 ESM 模型的映射算法，我们可以生成与图 1.2.2 中实体安全模型等价的 CPN 模型，如图 1.2.3 所示，图中的 CPN 模型中，颜色集、函数和变量的表示是基于 Standard ML 语言的[43]。入弧函数 $\text{in}(x)$ 对模型中不可信主体 $(L_{min}=L_{max})$ 和可信主体 $(L_{min}<L_{max})$ 对客体访问的安全规则 access 均给出了统一的准确描述；出弧函数 $\text{out}(x)$ 仅从位置 P 移出相应的 token，即 $(L_{max}, L_{cur}, \text{access})$。

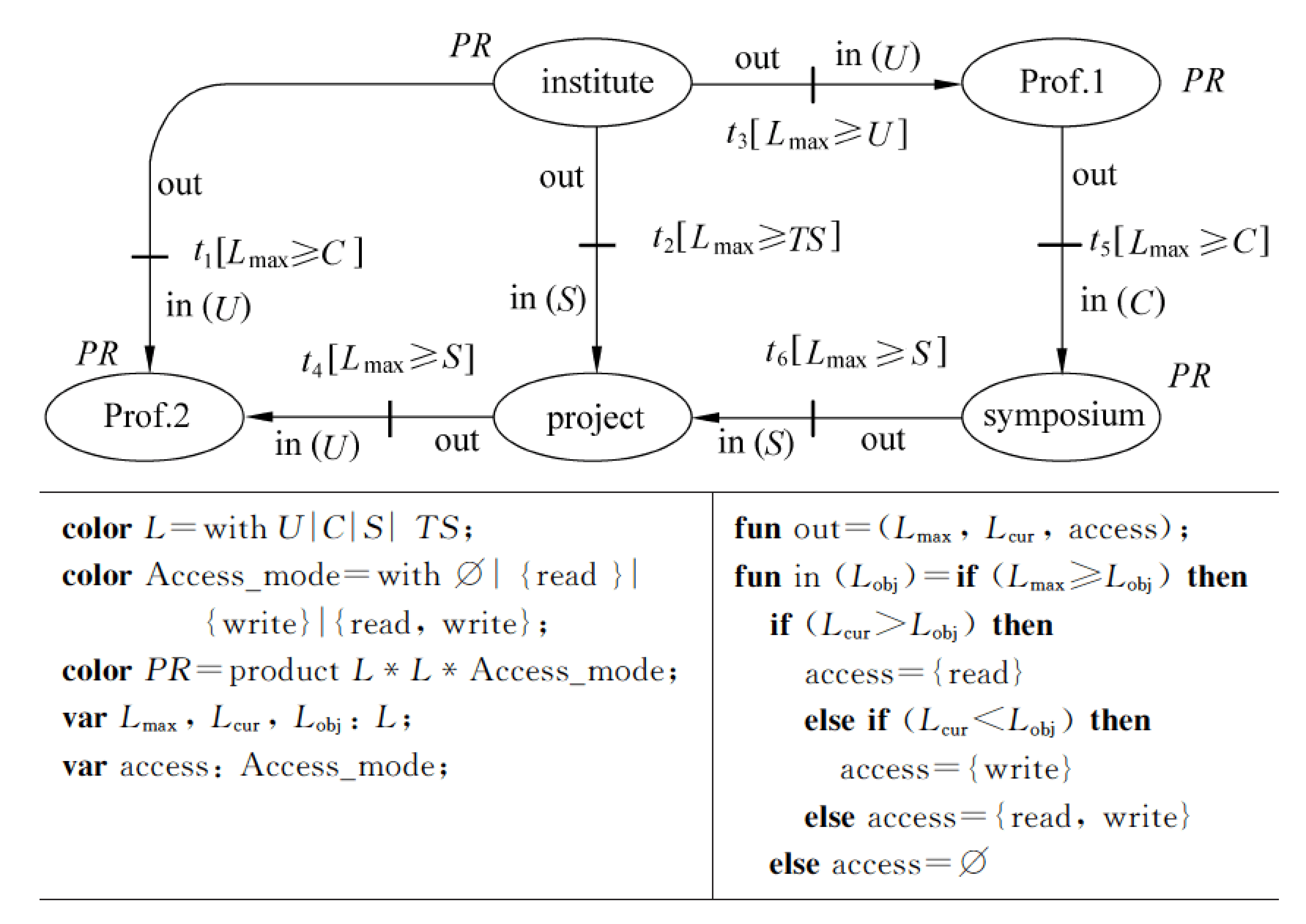


图 1.2.3　强制访问控制模型的等价 CPN 网模型

1.2.3.3　强制访问控制模型的访问可达图

设安全模型的 Petri 网系统的标识是函数 $M: P\to\{0,1\}$，即 $\forall p\in P, p\to\{0,1\}$。图 1.2.3 中的客体(institue，Prof.1，Prof.2，project，symposium)分别对应于 CPN 中的位置(P_1，P_2，P_3，P_4，P_5)，则标识映射可定义为：

(1) $m_i=M(p_i)=1$，当 p_i包含一个或多个 token，且 access$\neq\varnothing$；

(2) $m_i=M(p_i)=0$，当 p_i无 token，或者 access$=\varnothing$。

因此，基于 MAC 模型的 CPN 模型，可以构造主体对客体访问的状态可达图，并可利用现有的 Petri 网模型分析工具去验证系统的有关安全属性。

考虑生成图 1.2.3 的状态可达图，若考虑主体的动态安全级，则图 1.2.4 表示安全级为$[U, TS]$的可信主体($L_{\min}<L_{\max}$)的可达图；而图 1.2.5 则表示安全级为$[S, S]$的不可信主体($L_{\min}=L_{\max}$)的可达图。

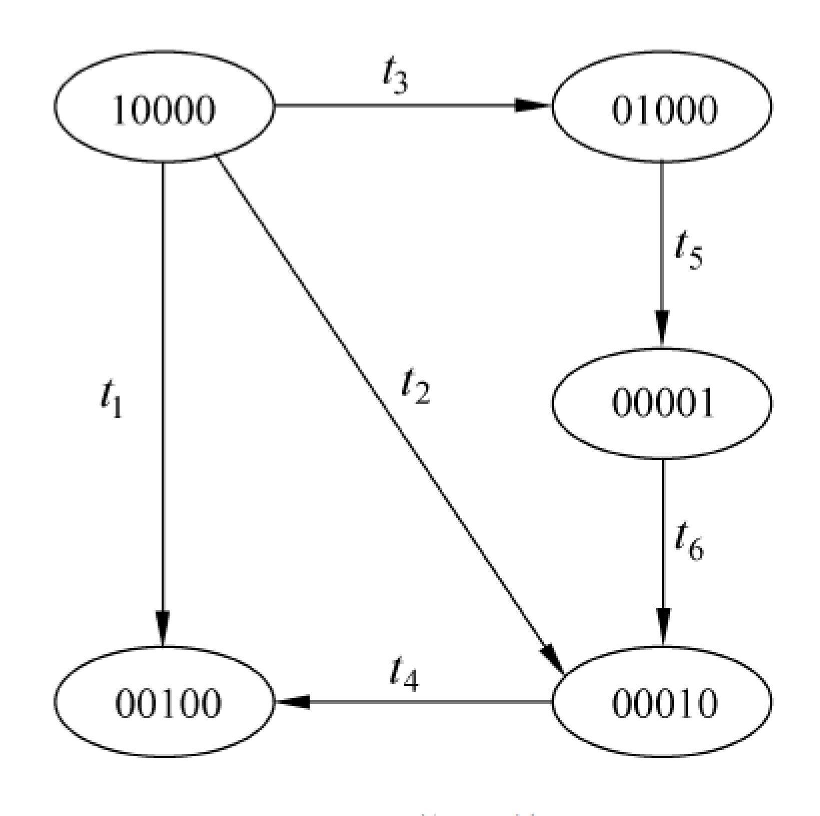


图 1.2.4　可信主体可达图

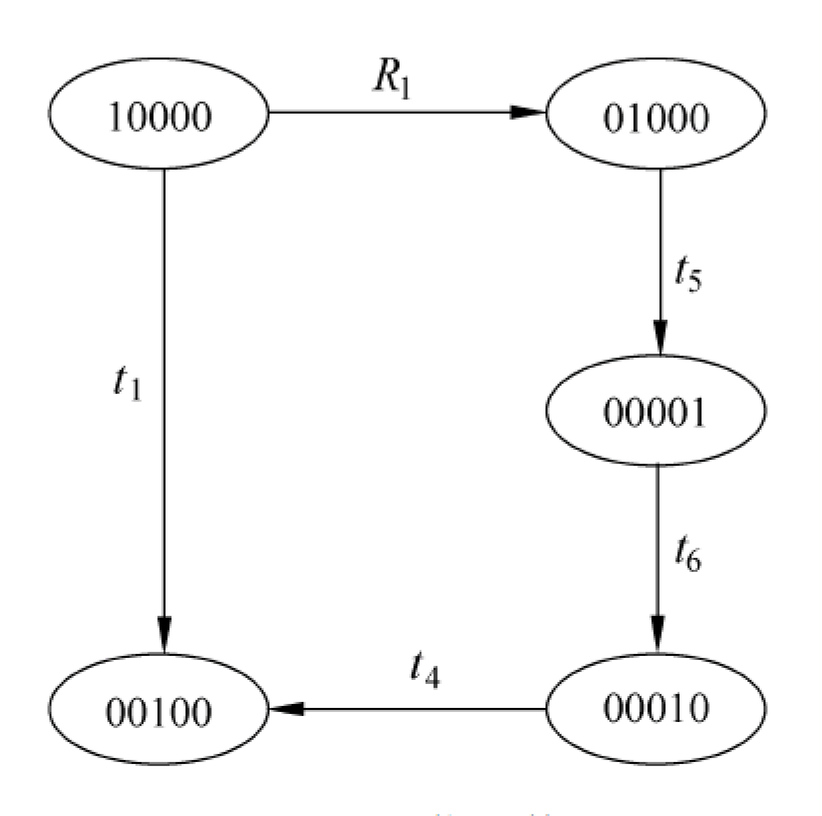


图 1.2.5　不可信主体可达图

值得指出的是，对于规模较小的系统，验证时模型的状态空间似乎并不是一个关键性的问题；然而，对于一个比较复杂的模型，CPN 的状态空间将随着实际系统的规模增长以指数形式呈爆炸性地增长。这种计算复杂性导致状态可达图分析方法在解决现实系统的分析问题中存在着某些困难，这一直也是形式化验证中一个值得研究的重要问题。

1.2.4　安全性分析

1.2.4.1　主体访问客体的时序特性

设客体 O_1 和客体 O_2 分别对应于 CPN 模型中的位置 P_1 和 P_2。基于安全模型的可达图，可以对主体访问客体的时序性质进行分析。考虑如下一些典型的时序性质：

性质 1.2.1（Follow 关系）　若主体在访问客体 O_1 后紧接着访问客体 O_2，则在可达图的每一条路径中，包含 P_1 的状态标识必须先于包含 P_2 的状态标识，即：

$$\text{Follow}(P_1, P_2) = \{(P_1, P_2) \mid (M_i(P_1)=1) \wedge (M_j(P_2)=1) \wedge (M_i[> M_j)\}$$ □

性质 1.2.2（Precede 关系）　Precede 关系与 Follow 定义相反：

$$\text{Precede}(P_1, P_2) = \text{Follow}(P_2, P_1)$$ □

性质 1.2.3（Adjacent 关系）　主体所访问的客体 O_1 和客体 O_2 是邻接关系，则在可达图的每一条路径中，包含 P_1 的状态标识和包含 P_2 的状态标识是邻接的，即：$\text{Adjacent}(P_1, P_2)=\{(P_1, P_2) \mid (M_i(P_1)=1) \wedge (M_j(P_2)=1) \wedge ((M_i[> M_j) \vee (M_j[> M_i))\}$。 □

性质 1.2.4（After 关系）　若主体对客体 O_1 访问先于对客体 O_2 的访问，且对客体 O_1 和 O_2 的访问不邻接，则在可达图的每一条路径中，包含 P_1 的状态标识必须要先于包含 P_2 的状态标识，即：$\text{After}(P_1, P_2) = \{(P_1, P_2) \mid (M_i(P_1)=1) \wedge (M_j(P_2)=1) \wedge (M_i[>M_{i+1}[>\cdots>M_j) \wedge (M_i \neq M_{i+1})\}$。 □

性质 1.2.5（Before 关系）　Before 关系的定义与 After 定义相反：

$$\text{Before}(P_1, P_2) = \text{After}(P_2, P_1)$$ □

性质 1.2.6（Mutex 关系）　若主体在访问客体 O_1 的过程中不能同时访问客体 O_2，则在可达图的每一条包含 P_1 的状态标识的路径中，不能出现包含 P_2 的状态标识，即：$\text{Mutex}(P_1, P_2)=\{(P_1, P_2) \mid (M_i(P_1)=1) \wedge \neg\text{After}(P_1, P_2) \wedge \neg\text{Follow}(P_1, P_2) \wedge \neg\text{Precede}(P_1, P_2) \wedge \neg\text{Before}(P_1, P_2)\}$。 □

对如上这些典型的时序性质，可以利用模型检测（model-checking）的方法对这些安全性质进行形式化的自动验证，相关的验证方法可参见文献[44,45]。

1.2.4.2　敏感信息的可推测性

如图 1.2.6 所示，设系统安全管理员授予了主体 S 对实体关系 R_2、R_3 和 R_4 的访问权限。尽管系统并未授权主体 S 对实体关系 R_1 的访问权限，但主体 S 还是可能推出在实体 E_1 和 E_2 之间存在敏感关联信息，如果在两个实体之间还存在其他路径：$E_1 \rightarrow E_3 \rightarrow E_4 \rightarrow E_2$。

若考虑图 1.2.2，设 Prof.1 参加一个关于某研究项目 Project 的会议 Symposium，他将可能推测出该研究所 Institute 所进行的研究项目，虽然系统管理员并未授予 Prof.1 对 Research 的访问权限。分析不可信主体的访问可达图(如图 1.2.5 所示)，我们可以注意到：尽管系统并未授予不可信主体(安全级为 S)对实体关系 Research(安全级为 TS)的访问权限，但是该主体还是有可能通过间接的访问路径知道研究所 Institute 的研究项目。这种安全隐患可以通过检测可达图来发现，因为在可达图中存在一条包含标识 M_i 和 M_j 的路径，且满足：

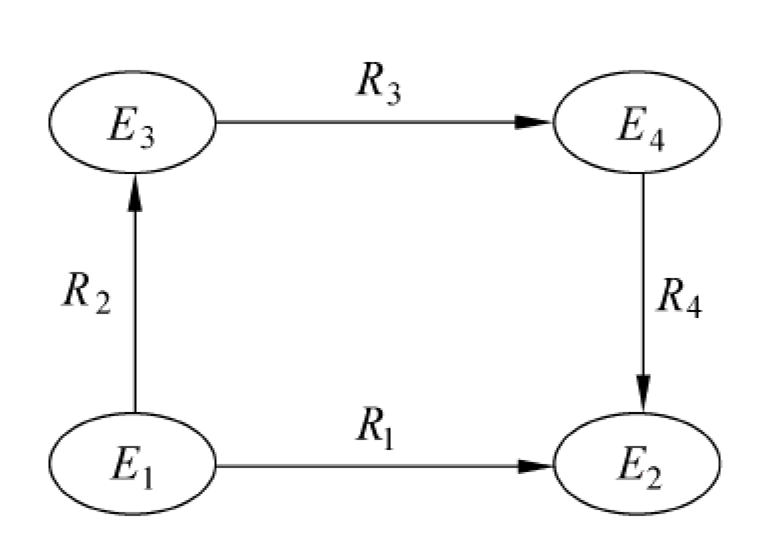


图 1.2.6 敏感信息的可推测性

$$(M_i(P_1)=1)\wedge(M_i(P_4)=0)\wedge(M_j(P_1)=0)\wedge(M_j(P_4)=1)$$

上式中，P_1 和 P_4 分别对应于实体 Institute 和 Project。

因此，根据以上分析可以得出如下更为一般性的结论：

定理 1.2.3 设 R 表示实体 P_u 和 P_v 之间的关系，且系统并未授予主体 S 对 R 的访问权限；若要防止主体 S 在访问实体 P_u 的过程中间接地推测出与实体 P_v 有关的敏感信息(实体间关系 R 的存在)，则在该主体访问的可达图中，对任意两个标识状态 M_i 和 M_j，必须满足如下条件：

$$(M_i(P_u)=1)\wedge(M_i(P_v)=0)\wedge$$
$$(M_j(P_u)=0)\wedge(M_j(P_v)=1)\rightarrow$$
$$\neg M_i[>M_{i+1}[>\cdots>M_j]\wedge(M_j\neq M_{i+1})]$$ □

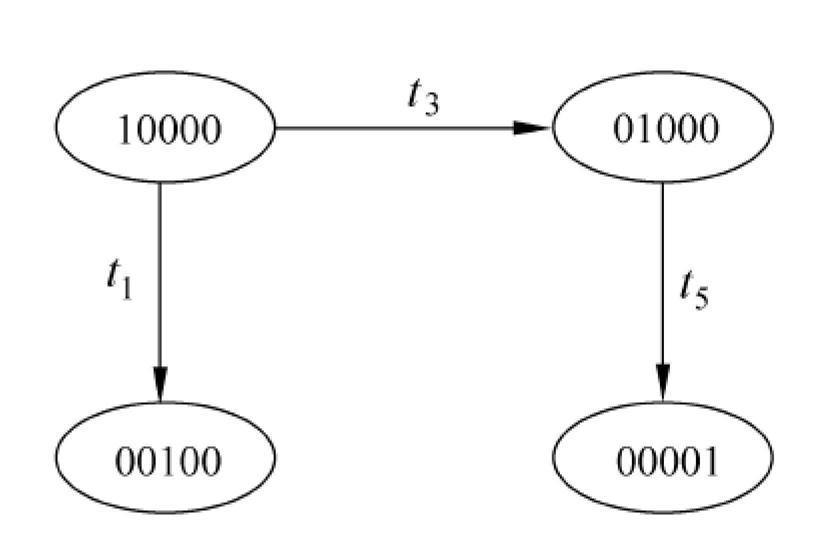


图 1.2.7 修改安全级的可达图

为了防止此类敏感信息的可推测性，若将实体间关系 Subject 的安全级 S 改变为 TS，则安全级为 S 的主体对客体 Project 的访问将不再可达，如图 1.2.7 所示。

1.2.4.3 主体动态安全级访问的安全隐患

在不允许动态改变主体安全级的情况下，强制访问控制模型中的安全策略对控制信息从低安全级流向高安全级是有效的。但是，使用静态安全级的策略势必影响到模型安全策略的灵活性，所以必须引入动态安全级的概念。然而，引入动态安全级，若不能对信息的流向进行有效的控制，将会造成潜在的安全隐患。

考虑如下情形：设主体 S 的最大安全级为 L_0，客体 O_1 和 O_2 的安全级分别为 L_1 与 L_2，且满足偏序关系 $L_0>L_1>L_2$。设在时刻 t_1，当主体 S 的当前安全级别为 L_0 时，因 $L_0>L_1$，所以 S 具有“读”O_1 的权限，这表明 S 从 O_1 读的信息 I；在时刻 t_2，当主体 S 的当前安全级别改变为 L_2 时，因主体的当前安全级和客体的安全级相等，同为 L_2，所以 S 具有“写”O_2 的权限，这表明 S 将信息 I“写”给 O_2。最终结果是：信息 I 从具有较高安全级别的客体 O_1 流向了较低安全级的客体 O_2，如图 1.2.8 所示，图中方括号内是实体的安全级。

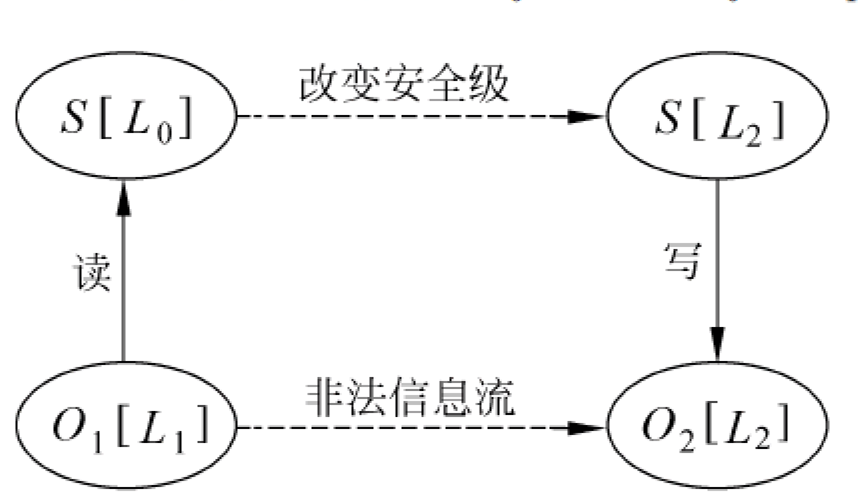


图 1.2.8 动态安全级访问的安全隐患

考虑图 1.2.4 中可信主体对客体的访问可达图。当可信主体访问客体 Project(安全级为 S)后，再对客体 Prof. 2(安全级为 U)进行访问，则将有可能出现客体 Project 中的信息 I 间接地流向客体 Prof. 2 的情况，而这种信息流向是非法的，因为信息 I 是从具有较高安全级别的客体流向了较低安全级的客体。

一种可行的检测算法是：在主体 S 的数据结构中同时保留有主体的最大安全级别 L_{max} 和当前安全级 L_{cur}，系统的安全管理员将对主体的当前安全级的变化进行跟踪。设系统记录主体 S 所“读”客体的最大安全级是 max，则根据我们定义安全级的格模型和安全策略，这意味着有信息流 I：max→L_{max}。另外，若在某时刻主体 S 要将其当前安全级更改为某个客体所拥有的安全级 min 时，这意味着有信息流 I：L_{max}→min。由信息流 I：max→L_{max} 和 I：L_{max}→min 可推出隐含的信息流 I：max→min。信息流的合法性可由系统安全管理员来判定：当 min≤max 时，这是合法的信息流；当 max≤min 时，这是非法的信息流。因此，根据如上检测算法可得出如下结论。

定理 1.2.4　设 CPN 模型中位置 p 和 q 分别表示强制访问控制模型中的两个客体，其相应的安全级是 $f(p)$和 $f(q)$。若在访问客体的过程中主体的安全级动态可变，则为了防止在主体访问不同客体时出现非法的信息流，应满足如下约束条件：

$$\forall p, q \in P,\quad M_i[> M_{i+1},\ (M_i(p) = 1) \land (M_{i+1}(q) = 1) \rightarrow f(p) \leqslant f(q)$$ □

从主体访问的状态可达图中标识变迁的定义以及强制访问控制模型中主体对客体的强制访问策略可知，上述结论显然是成立的。

1.2.4.4　主体对客体的访问可达性

标识可看成给定时刻的系统状态，可达图中的每个节点对应于每个标识，从根标识节点开始的可达图反映了系统从初始状态开始后的状态变化。可达图的路径可定义为标识序列 $M_0, M_1, \cdots, M_n$，其中，对于 $\forall i < j\ (i, j=1,2,\cdots,n)$，存在变迁序列 $Y= t_1 t_2 \cdots t_n$ 使得从标识 M_i 可以到达标识 M_j，记为 $M_i[Y>M_j$。

定义 1.2.13　若主体被授权能够访问某个客体，并且存在一条路径，主体通过该路径能够访问到该客体，则表明该客体是访问可达的；否则，若主体被授权能够访问某个客体，但不存在这样一条访问可达路径，则称该客体访问不可达。□

定理 1.2.5　设 CPN 模型中位置 P_k 表示强制访问控制模型中的某个客体，则在主体的访问可达图中，若该客体对该主体而言是访问可达的，则至少存在一个标识 M_i 和变迁序列 Y 满足条件：$(M_i(P_k)=1) \land (M_0[Y>M_i)$。□

考虑图 1.2.7，在将实体间关系 Subject 的安全级 S 改变为 TS 后，尽管系统安全管理员允许主体访问客体 Project(主体的安全级等于客体的安全级 S)，但是安全级别为 S 的主体对客体 Project 的访问将不再可达。

1.3　支持移动通信的访问控制

随着信息技术和通信技术的发展，人们的通信方式发生了日新月异的变化，移动通信为人们提供了一个完整的可靠传输方式，使用户摆脱终端的束缚，实现任何时间和地点之间的

通信。此外，Internet 的一个发展趋势是支持移动无线网络，移动 IP[46,47] 在这种背景下应运而生，能够保证一个无线网络节点从网络连接的一端自由地移动到另一端，而不会中断端到端的网络连接。

IPv6 是下一代互联网协议，具有很好的扩展性和兼容性，它最终将代替 IPv4 成为互联网的主要网络协议。本节基于层次移动 IPv6(hierarchical mobile IPv6，HMIPv6)[48] 结构，介绍了一种有效结合访问控制和移动管理的方法，其思想是通过 HMIPv6 的本地移动管理改善认证延迟和带宽消耗，并设计了优化方法以加速访问认证点获得移动节点的认证信息。

1.3.1 移动 IPv6

移动通信起源于 20 世纪 70 年代，经过几十年的发展，现代移动通信经历了从第一代模拟的移动通信系统，到 20 世纪 90 年代数字化的 GSM 系统，再到当前第三代移动通信系统的发展变革。移动通信系统的发展为用户提供了更多的互联网业务。在当前移动通信网的标准中，IP 技术已经得到充分的应用，要求每个移动终端都是 IP 可达的，且至少拥有一个全球唯一的 IP 地址。移动 IP 允许移动节点从一个链路移动到另一个链路而不需要改变移动节点的 IP 地址。

在移动通信网络中，无论移动节点移动到网络中的任何地方，目的地址是移动节点家乡地址的包都能够路由到移动节点，且移动节点移动到新链路后可以继续与其他节点保持通信，保证了移动节点的移动对于传输层或更高层协议和应用的透明性。而 IPv4 对移动互联网具有很多限制，因此只有 IPv6 才能满足这种需求，移动通信的发展迫切需要 IPv6。

IPv6 与移动通信的结合将为目前的互联网开拓一个全新的领域，无线网络将成为 IPv6 的一个重要应用，且是实现移动互联网上服务的关键。IPv6 的出现是移动计算的一个重要里程碑，其主要特性对于未来的移动无线网络的发展至关重要，这些特性包括：足够多的 IP 地址、安全数据包头的实现、目的选项提高了路由效率、地址自动配置、避免入口过滤等。

移动 IPv6(mobile IPv6，MIPv6)允许 IPv6 主机在离开其家乡网络后仍能很好地维持其原有的通信，并与 Internet 很好地连接。移动 IPv6 的实现机制对 TCP、UDP、应用层等都是透明的，对移动用户来说，丝毫不会感觉到其移动对通信的影响。此外，IPSec 是集成在 IPv6 中强制实现的，因此移动 IPv6 能够提供比移动 IPv4 更好的安全性。移动 IPv6 使用户能在同类或不同类的网络中进行漫游和无缝的切换，而对移动性的支持是 IPv6 最显著的特点。随着无线通信技术和互联网的发展，越来越多的用户成为移动用户，由于 IPv6 协议中的邻居发现协议、无状态地址自动配置协议、IPv6 封装及路由优化等特点，IPv6 不仅能够支持大量的移动用户，且对移动用户的移动管理也更加简单、有效。

移动 IPv6 由以下几个部分组成：

(1) 移动节点(mobile node，MN)：可更改链路并使用其家乡地址 (home address) 保持可连接性的 IPv6 节点。

(2) 家乡链路(home link)：生成移动节点的链路。

(3) 家乡地址(home of address)：分配给连接到家乡链路的移动节点的地址，而且通过该地址始终可以访问相应的移动节点，无论其在 IPv6 网络上位于何处。由于家乡地址总是分配给移动节点，因此移动节点在逻辑上总是连接到家乡链路。

(4) 家乡代理(home agent,HA):家乡链路上的一台路由器,保存离开家乡地址的移动节点的注册信息及其当前地址。虽然家乡代理充当将家乡链路连接到 IPv6 网络的路由器,但是家乡代理不是必须提供这项功能,家乡代理也可以是家乡链路上的一个节点。

(5) 外地链路(foreign link):不属于移动节点的家乡链路的链路。

(6) 转交地址(care-of address,CoA):移动节点在连接到外地链路时所用的地址,移动节点的家乡地址与转交地址的关联称为绑定。

(7) 通信节点(correspondent node,CN):与移动节点通信的 IPv6 节点。通信节点不一定必须支持移动 IPv6。

1.3.2　支持移动网络的访问控制

为了扩展移动 IP 的应用领域,需要考虑其他一些机制,以保证访问域代理能够验证移动节点的身份,并基于本地策略和合同策略与远端的家乡域代理授权连接。然而,现有的访问控制机制多数是为固定网络设计的,这就对支持移动性的情况提出了挑战[49]。例如,接入网的认证点并不能拥有移动节点的认证信息,如身份、计费、服务的级别等,这样将会导致阻止认证决策,中断请求甚至丢失请求。即使接入网中的认证点能够得到移动节点的认证信息,为了提供有效的移动管理需要考虑其他一些重要问题,如怎样降低由于交换认证信息而引起的延迟和带宽消耗。当移动节点进入到一个新的子网时,由于临时地址的变化以及缺少认证信息,将会初始化一个新的证明和认证程序,这样会限制移动节点通知家乡代理和通信节点它目前的位置,直到证明完成。此外,移动 IP 仅能够解决移动节点的 IP 路由,并用一些安全机制来保护位置管理消息,如 IPSec。

基于下一代 Internet 协议,移动 IPv6(MIPv6)显示出一些新的重要特征,使其比 MIPv4 更易于被接受并广泛应用。为了跟踪移动节点,MIPv6 要求一个移动节点汇报与当前接入网相关的地址给家乡代理和一些通信节点,而优化路由是移动 IPv6 的一个显著特征。访问控制能够限制用户访问一些信息,并可以根据用户身份和预先定义的规则实施一些功能,一般情况下,访问控制由系统管理员制定,控制用户访问服务、文件和网络资源等。我们将移动网络下的访问控制分为两种类型,如图 1.3.1 所示。

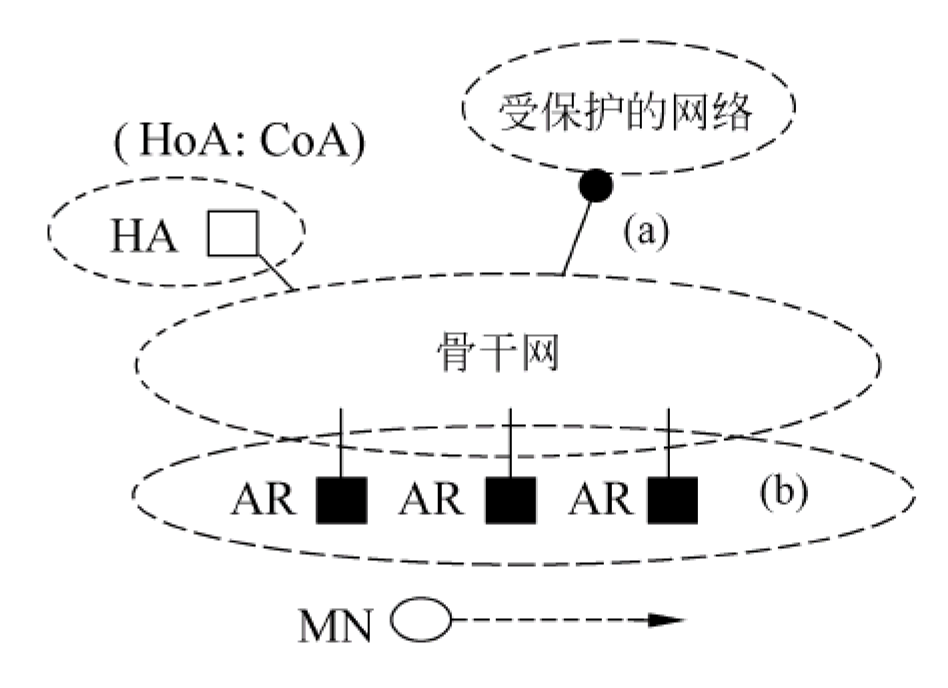


图 1.3.1　移动 Internet 环境下的访问控制

类型(a)由一个单一的认证点保护资源,该认证点存储了认证信息并检测所有静态和移动节点的访问请求,因此,由于存在大量的转交地址(care-of address,CoA),支持移动的访问控制主要是认证移动节点(mobile node,MN)。为了解决这个问题,可以采取用户名的认证。在类型(b)中,多个认证点与控制移动节点交互以访问接入网的资源,如访问服务。支持移动网络的访问控制过程如下:

(1) 由于存在大量的转交地址,移动节点的身份必须通过认证;

(2) 由于每个认证点无法存储所有移动节点的认证信息,如移动节点在移动过程中的访问路由器(access router,AR),因此有必要交换这些认证信息,但是这样会消耗网络带宽。

此外，等待认证信息又会增加一定的延迟，很难保证信息交换时的安全性。

实质上，控制移动节点访问路由是一种很重要的访问控制机制，因此本节的重点是第二类访问控制，下面提出的方法能够减少信息交换的代价并加速进程的执行。

1.3.3 支持层次移动 IPv6 的访问控制

1.3.3.1 层次移动 IPv6

层次移动 IPv6 的目的是：通过采用层次型路由器结构，减少移动节点与家乡代理和通信对端的信令交互，减少切换引起通信中断的时间。HMIPv6 的相关术语如下：

(1) 移动锚点(mobile anchor point，MAP)：是一个处于移动节点所访问网络上的路由器，相当于移动节点的本地家乡代理，类似于移动 IPv6 中的家乡代理。

(2) 访问路由器(access router，AR)：移动节点当前的接入路由器。

(3) 区域转交地址(regional care-of address，RCoA)：移动节点从所访问的网络获得的转交地址，是基于 MAP 子网前缀的地址，根据移动节点收到的 MAP 选项的内容自动配置，只要移动节点在同一个 MAP 子网内，移动节点的区域转交地址就不会改变。

(4) 链路转交地址(on-link CoA，LCoA)：区别于区域转交地址，它是基于移动节点当前的默认路由器前缀和移动节点的接口标识形成的转交地址，它标识移动节点当前的确切位置。

(5) 本地绑定更新(local binding update)：移动节点向 MAP 发送本地绑定更新，用来在 MAP 域里建立起区域转交地址(RCoA)和链路转交地址(LCoA)之间的绑定关系。

在 HMIPv6 中，移动锚点(MAP)可以是 HMIPv6 网络中任何层次的路由器，MAP 可以限制移动 IPv6 与本地域以外节点的信令交互，能够支持快速移动 IP 切换，帮助移动主体实现无缝移动，并且支持特定的移动网络情况。移动主机通过 MAP 获得的地址是区域转交地址 RCoA。根据 RCoA，MAP 有两种模式：基本模式和扩展模式。当一个移动节点漫游到 MAP 域时，移动主机可以在 MAP 的子网上形成自己的 RCoA，称为基本模式，该模式只是对移动节点的操作进行扩展，而没有对家乡带理和通信节点的操作进行任何改动；或者移动主机使用 RCoA 作为备用的转交地址，称为扩展模式，该模式对移动主机和家乡代理的操作进行了少量扩展，而没有对通信节点的操作进行修改。

HMIPv6 将一些路由器配置在 MAP 上，移动节点 MN 将这些移动锚点作为本地家乡代理，从而减少了骨干网中本地注册消息的数量。首先，移动节点察看路由器通告消息(router advertisement messages)以检测 MAP 域的变化，并用无状态自动配置方法分别配置 LCoA 和 RCoA[50,51]及 MAP 选项；MN 要求 MAP 在 RCoA 和 LCoA 之间建立一个绑定实体，然后 MN 通知 CN 和家乡代理 HA 在 RCoA 和 HoA(home of address，家乡地址)之间建立一个绑定实体，如果 MN 在本地 MAP 中改变了当前地址，它仅需要在 MAP 处注册新地址即可。因此，只有 RCoA 需要在 CN 和 HA 中注册，只要 MN 在 MAP 域内移动，RCoA 就不会改变。这样就使得节点的移动性对 CN 来说是透明的，由于 MAP 能够处理 MAP 域内移动节点的本地注册消息，极大地减少了通信的代价。

图 1.3.2 是 CN 发送数据包时的路由传输过程。MAP 可以帮助移动节点 MN 在与

CN 通信时无缝地在 AR 中移动。当 MN 到达外地网络时，MN 发现 MAP 的全局地址，并存入 AR，通过路由器通告消息发送到 MN。通常数据包被发送到家乡网络，中途被 HA 获得并封装成 RCoA，MAP 使用代理邻居通告(proxy neighbor advertisement)对 RCoA 中的数据包进行解释，然后数据包被封装并路由到 MN 的 LCoA。当 CN 发送的数据包在本地成功注册后，通过路由扩展头进入 RCoA，并由 MAP 以隧道方式转发到 LCoA。在反方向上，MN 通过家乡地址目标项将数据包发送到 CN。

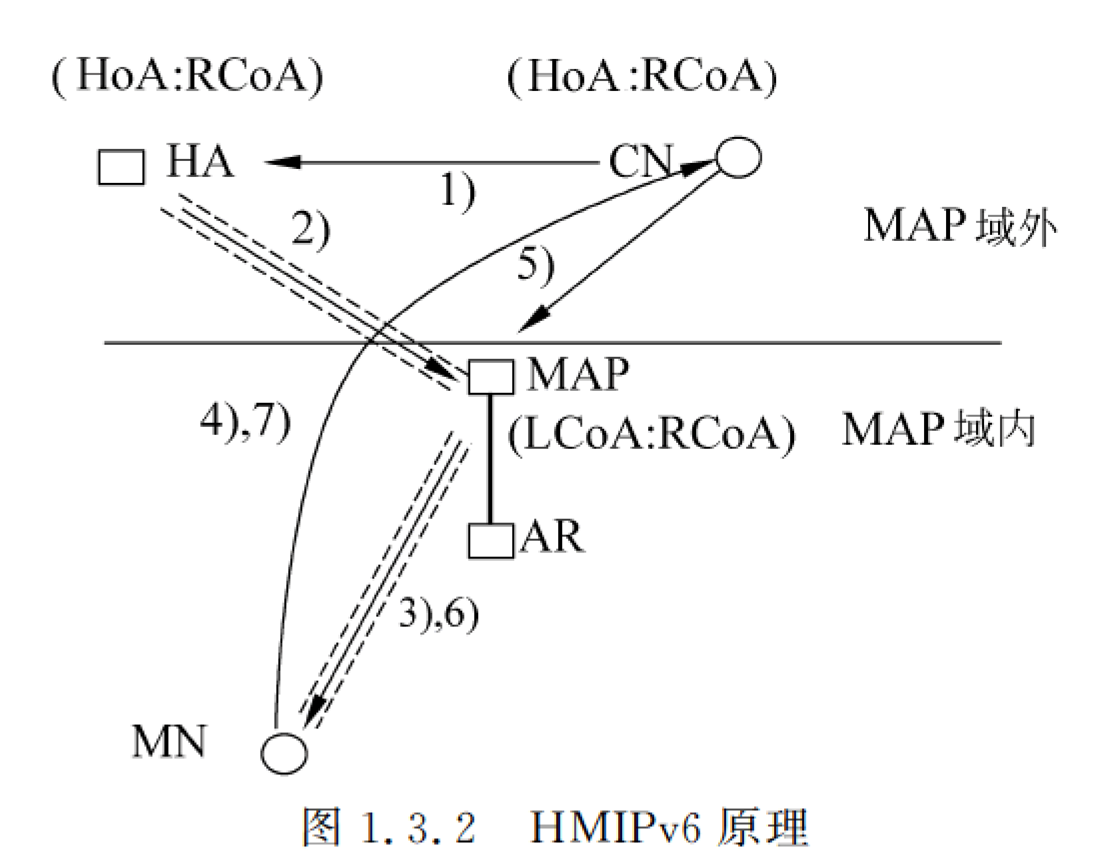


图 1.3.2　HMIPv6 原理

1.3.3.2　支持 HMIPv6 的访问控制框架

图 1.3.3 是一种有效支持 IPv6 网络移动性的访问控制方法，该机制基于层次移动 IPv6。服务器 F_server 作为验证 MN 身份的代理并授权每个 MAP 域。在 MN 的家乡网络中，服务器为 H_server，F_server 通过询问 H_server 以获得本地 MAP 域中 MN 的认证信息。这里需要一种机制来保证 F_server 和 H_server 信息存储的同步性。需要注意的是，这两种服务器在网络拓扑中是逻辑配置的。

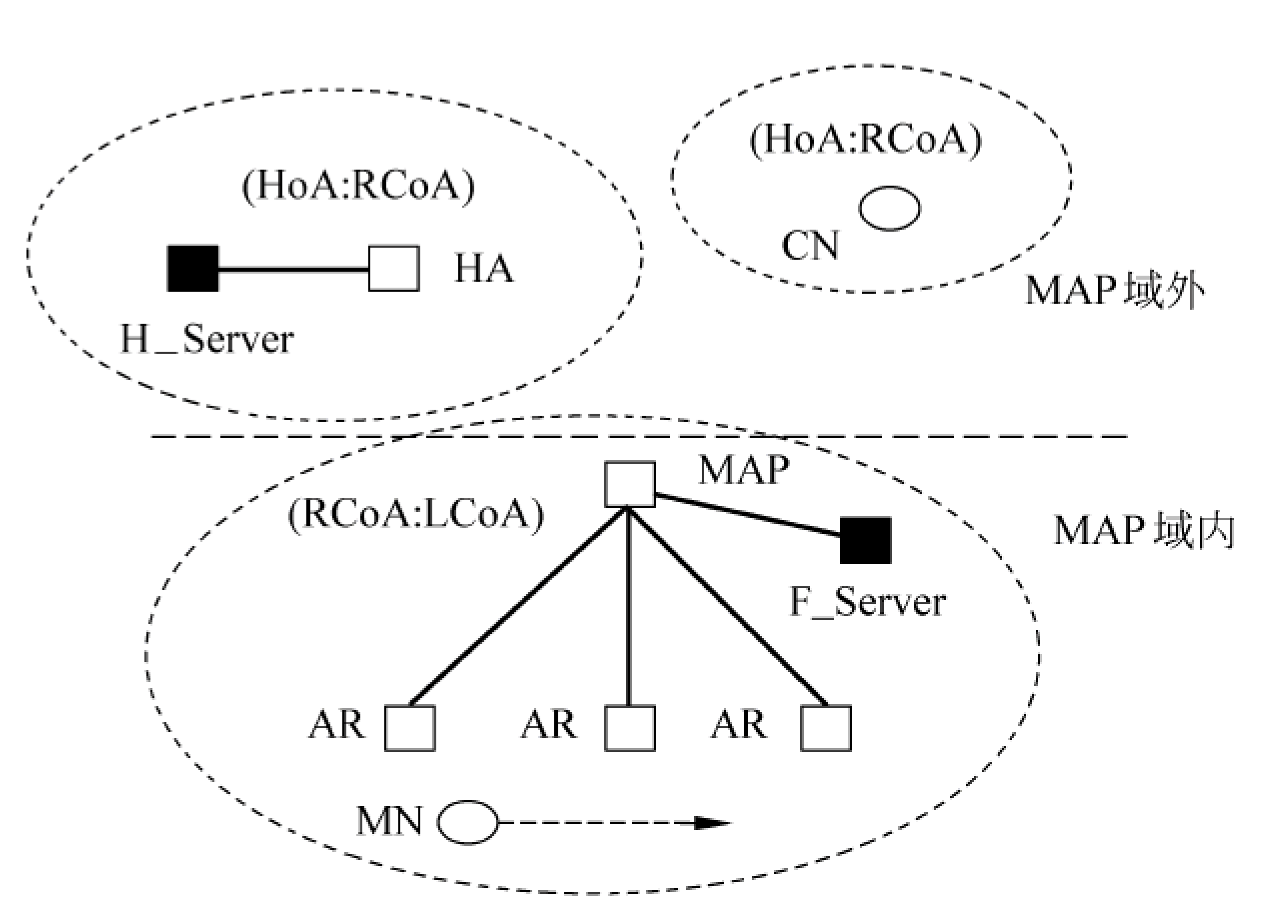


图 1.3.3　支持层次移动 IPv6 的访问控制框架

当 MN 进入 MAP 域内时，AR 通过咨询 F_server 控制 MN 对网络的访问。另一方面，数据包发送到 MAP 域内后被 MAP 捕获，将数据包解封并询问 F_server 是否转发或丢弃该数据包。

通过这两个服务器可以看到，当 MN 漫游在一个 MAP 域内时，RCoA 作为 MN 的信标是保持不变的。当 MN 第一次进入到 MAP 域内时，自动配置协议会给 MN 初始一个 RCoA 和 LCoA，然后 MAP 触发 F_server 询问 MN 所在 HA 域内的 H_server，以获得用户的信息，然后 MN 在 HA 和 MAP 中进行本地绑定或更新。当 MN 的 AR 发生变化，且只有当一些信息被更新时，F_server 和 H_server 之间的协商才能够执行，这样就极大地降低了由移动管理访问控制所引起的代价。

考虑到 HMIPv6 的层次性，这里提出了两层访问控制结构。对于底层访问控制，根据本地策略对 MN 进行认证，以授权 MN 访问 MAP 域的资源。对于上层访问控制，根据 H_server 的本地策略和远程策略，通过认证后的 MN 可以访问 MAP 域以外的资源。在底层访问控制中，F_server 作为一个中心节点管理所有 MAP 域内的移动节点，并响应 AR 的请求，允许 AR 独立地执行访问控制。这里主要介绍上层访问控制。

1.3.3.3 数据结构

除了认证信息外，我们需要对 H_server 和 F_server 这两个数据结构进行介绍，如图 1.3.4 所示。在 F_server 的数据结构中需要记录移动到该域的 MN 的 H_server 和时间戳，时间戳的功能是保持认证信息被周期性地更新。H_server 的数据结构与 F_server 的类似，很明显，F_server 的时间戳低于 H_server 的时间戳。过期的时间戳将会从 F_server 中删除。

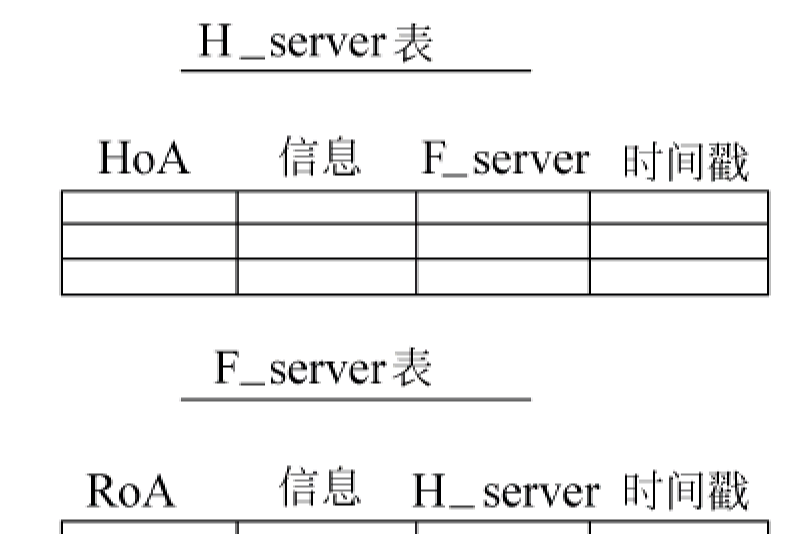


图 1.3.4　基本的数据结构

获得 H_server 和 F_server 地址的方法是修改 MAP 和 HA 的协议进程。当 MN 进入 MAP 域时，MAP 向 HA 发送请求并询问 MN 的认证信息，HA 将询问转发到 H_server，询问的内容包括 F_server 的地址和 MN 的身份。MAP 通过本地位置注册或 F_server 的间接通知来检测新来的 MN，但是 F_server 的间接通知并不能获得任何 MN 的认证信息。MAP 获得 MN 的 HA 地址是比较困难的，虽然获得 HA 地址并不能有效地分析 MN 发送数据包的协议结构，但是由于仅当有新来的 MN 时，进程才被执行，因此这里将采用这种方法。

1.3.3.4 协议步

假设在本地访问控制系统中，MN 能够使用 MAP 域内的资源，其关键协议步如图 1.3.5 所示。

Step 1. MN 通过分析信标消息执行移动检测，如由 AR 发送的路由器通告。当 MN 进入一个新的 MAP 域内时，MN 通过自动配置协议获得两个新地址：LCoA 和 RCoA。

Step 2. MN 分别与 MAP 和 HA 启动本地和全局位置注册(location registration)进程。如果访问控制系统正在运行，则全局位置注册可能会失败。通过分组域，MAP 获得 MN 的 HA 地址。

Step 3. MAP 向 F_server 发送请求，询问 MN 的认证信息。如果没有获得有用的信息，则 MAP 认为 MN 是一个新加入的节点，也可以通过本地绑定缓存查询 MN 的认证

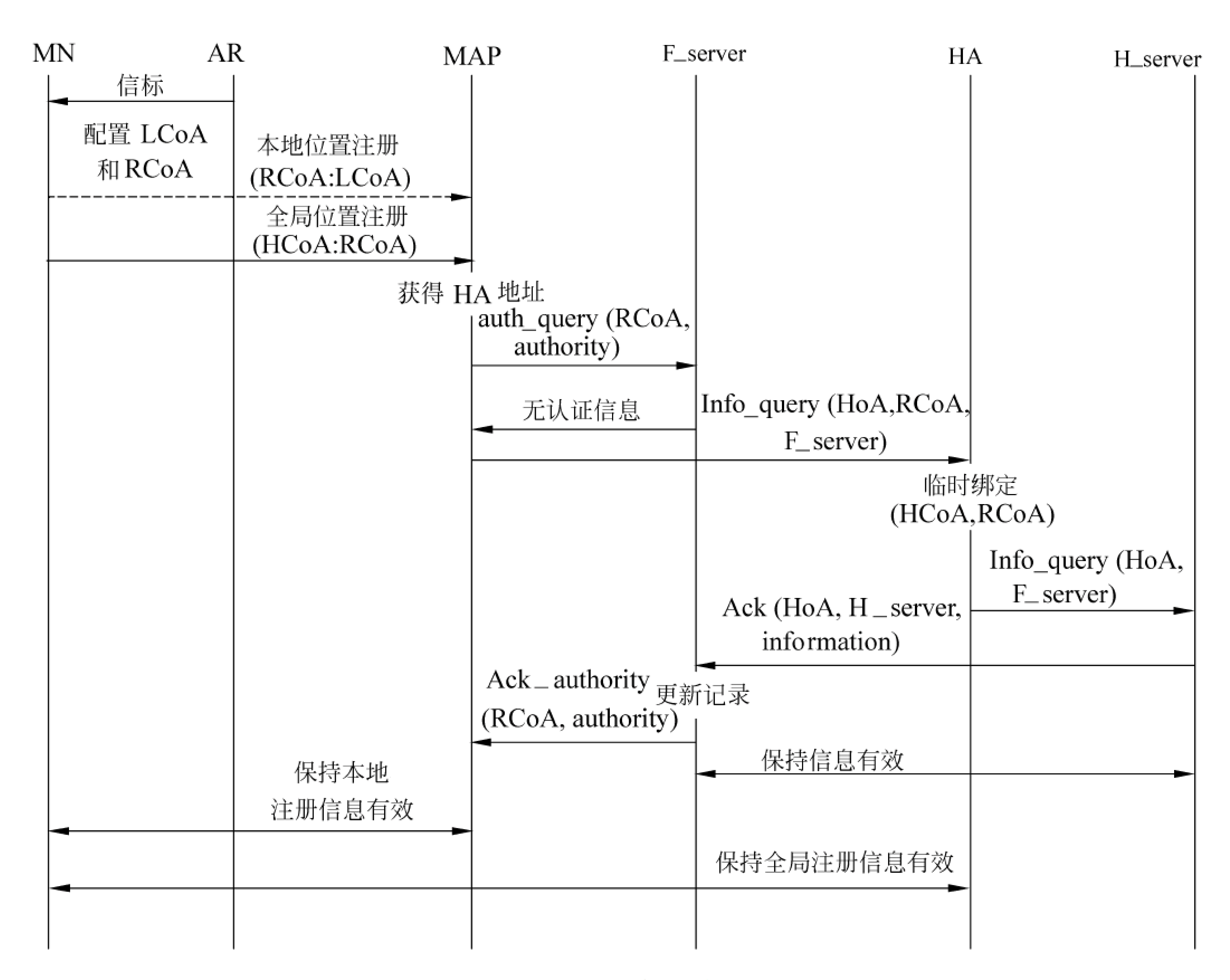


图 1.3.5　协议步

信息。

Step 4. MAP 向 HA 发送请求，询问 MN 认证信息，该信息包括 HoA，RCoA 和 F_server 的地址。

Step 5. HA 在 HCoA 和 RCoA 之间建立一个临时绑定项，即 MAP 使用全局位置注册来代表 MN，以减少交换认证信息引起的延迟。

Step 6. HA 转发请求给 H_server。

Step 7. H_server 向 F_server 发送 MN 的认证信息作为回答。

Step 8. F_server 更新自己的记录，并通知 MAP 和通信节点执行访问控制。

Step 9. F_server 和 H_server 直接交换认证信息以保证信息的有效性。

Step 10. MN 周期性地向 MAP 汇报自己的本地位置，更新全局位置并直接绑定在 HA 上。

1.3.4　方案的扩展与分析

如上文所述，当将访问控制系统和移动管理相结合时，减少网络的带宽和切换延迟是很重要的。在本节提出的方法中，通过 HMIPv6 的本地移动管理减少了交换 MN 认证信息的数量。同时，MAP 可以在 HA 建立一个临时位置绑定来代表 MN，这样，当 MN 的 MAP 域

发生改变时,更新 HA 绑定的延迟会大大改善。

然而,等待认证信息又增加了在 CN 绑定的延迟时间。为了减少 MAP 域之间的切换时间,我们需要一种新的访问控制方法,这里采取用户信息中转的方式,如图 1.3.6 所示,新的 F_server 能够从邻近的旧 F_server 中获得 MN 的信息。为了实现这种机制,一种方法是当前的 F_server 请求旧 F_server 转发新加入移动节点的认证信息,但是如果不更改移动节点的协议栈,当前 F_server 很难获得旧 F_server 的地址。另一种简单的方法是在 MAP 切换时间增加之前,让 F_server 通告新加入 MN 的认证信息到邻近的 F_server,这里需要一些移动预报算法以限制通告消息的数量[52]。

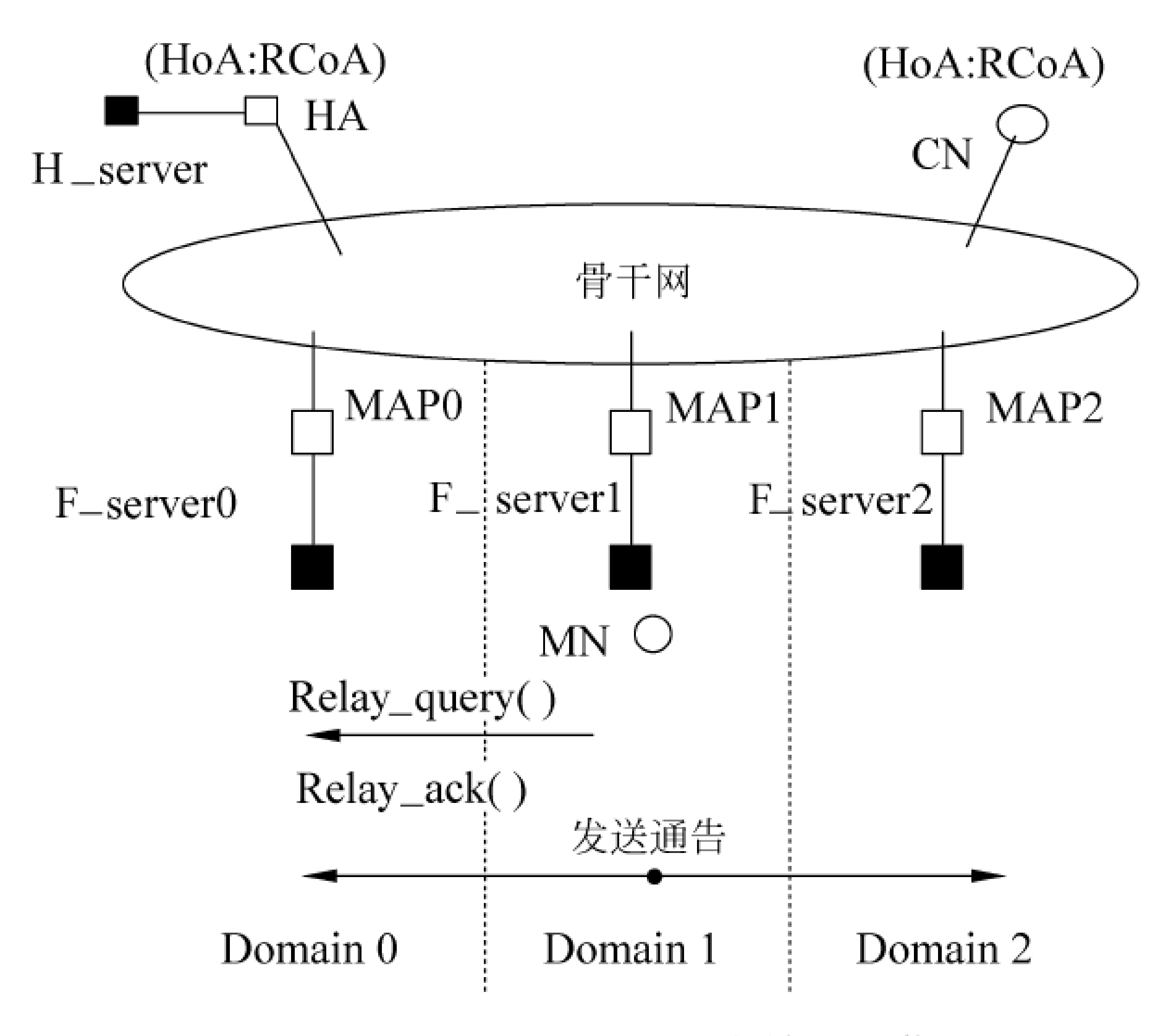


图 1.3.6 通过旧 F_server 中转用户信息

用户信息的安全性也是很重要的,在 1.3.3 节介绍的方法中,F_server 需要从 H_server 中获得认证信息,这里必须确保保密性、完整性和有效性。与移动 IP 类似,IPSec[53]可以用来在 F_server 和 H_server 之间建立一个安全通道。由于 F_server 和 H_server 被定义为逻辑实体,因此可以安装在适当的节点中。如果将 MAP 和 HA 分别配置成 F_server 和 H_server,记录缓存可用来保存记忆,这样就会降低建立安全通道的复杂性。

基于 HMIPv6 结构的访问控制方法能够支持 IPv6 网络的移动性,F_server 和 H_server 的引入减少了切换时间和管理的代价。此外,可以对该方法进行改进,如在本地中转用户信息,这样可以进一步提高通信效率,或设计一些具体的协议,以保证 F_server 和 H_server 之间能够安全地交换信息。

1.4 可信网络访问控制与可信网络连接

随着计算技术的普及,计算机系统应用日益广泛。由于计算机系统处理的任务多样化,计算机系统的工作环境普适化,计算机系统也越来越复杂,面临的各种人为和非人为的威胁也越来越多,如恶意攻击、垃圾邮件、计算机病毒等,导致人们对网络的不信任。计算机系统

能否正确、安全、高效地完成指定任务，即能否提供可信赖的服务能力，将成为研究人员关心的主要问题之一。计算机系统这种提供可信赖服务的能力就是可信性，如何确保计算机网络的可信性是未来计算新的研究方向。可信网络是可信计算发展的必然趋势，是下一代互联网发展的必然目标[54]。

1.4.1 可信网络

可信网络借鉴了系统可信性的概念，将传统孤立的研究内容融合到网络可信这一目标下，面向用户提供系统的安全服务。

可信性（trustworthiness）比安全性更富有广泛的技术内涵，在一定程度上前者比后者更为重要。这是信息安全研究领域近来取得的一个新共识。可信的互联网络应该具有如下特性：

（1）实现传统意义上的安全性，即系统和信息的保密性（confidentiality）、完整性（integrity）、可用性（availability）；

（2）真实性（authenticity），即用户身份、信息来源、信息内容的真实性；

（3）可审计性（accountability），即网络实体发起的任何行为都可追踪到实体本身；

（4）私密性（privacy），即用户的隐私是受到保护的，某些应用是可匿名的；

（5）可生存性（survivability），在系统故障、恶意攻击的环境中，能够提供有效的服务；

（6）可控性（controllability），是指对违反网络安全政策（security policy）的行为具有控制能力。

从理论上讲，根本上消除脆弱性、企图设计并实现一个绝对安全的互联网络是不切实际的。但新一代互联网络需要从体系结构上为可信性付出必要的努力，至少将安全完全建立在对用户绝对信任基础上的假设是不能再成立的。

1.4.1.1 可信网络概述

自 20 世纪 60 年代以来，互联网络在规模和应用领域上日益得到拓展，随着传感器、嵌入式设备、消费电子等设施的大量接入，网络的规模仍在继续膨胀。我国电信、金融、科教、交通等网络蓬勃发展，尤其是近年来电子商务和电子政务等网络应用的出现，突破了传统领域的网络应用形式，网络在国民经济生活中的基础性、全局性作用日益增强。尽管互联网已经转变并极大地改善了人类社会经济生活的方式，但同时也不得不面临大量的网络安全问题，如恶意攻击、垃圾邮件、计算机病毒、不健康资讯等，这些都导致人们对网络的不信任。

根据中国互联网信息中心（CNNIC）2008 年 1 月公布的第 21 次《中国互联网络发展状况统计报告》：截至 2007 年 12 月，网民数已增至 2.1 亿人。中国网民数增长迅速，与 2007 年 6 月相比增加了 4800 万人，2007 年一年则增加了 7300 万人，年增长率达到 53.3%。IP 地址和域名是互联网的基础地址资源，年增长率分别达到了 38% 和 190.4%，保证了互联网发展的平稳进行。CN 域名数 2007 年一年增加了 4 倍。网站数、网页数和网页字节数超过 60%的增长速度，反映了网上信息资源的增加速度很快，网民可以享用的信息资源越来越丰富。

根据国家计算机网络应急技术处理协调中心（CNCERT/CC）2007 年网络安全工作报

告,2007 年,我国公共互联网整体上运行基本正常,但从 CNCERT/CC 接受和监测的各类网络安全事件情况可以看出,网络信息系统存在的安全漏洞和隐患层出不穷,网络攻击的种类和数量成倍增长,基础网络和重要信息系统面临着严峻的安全威胁。2007 年网络安全事件主要有网络仿冒、垃圾邮件和网页恶意代码事件等,根据报告的事件类型的统计情况如图 1.4.1 所示。

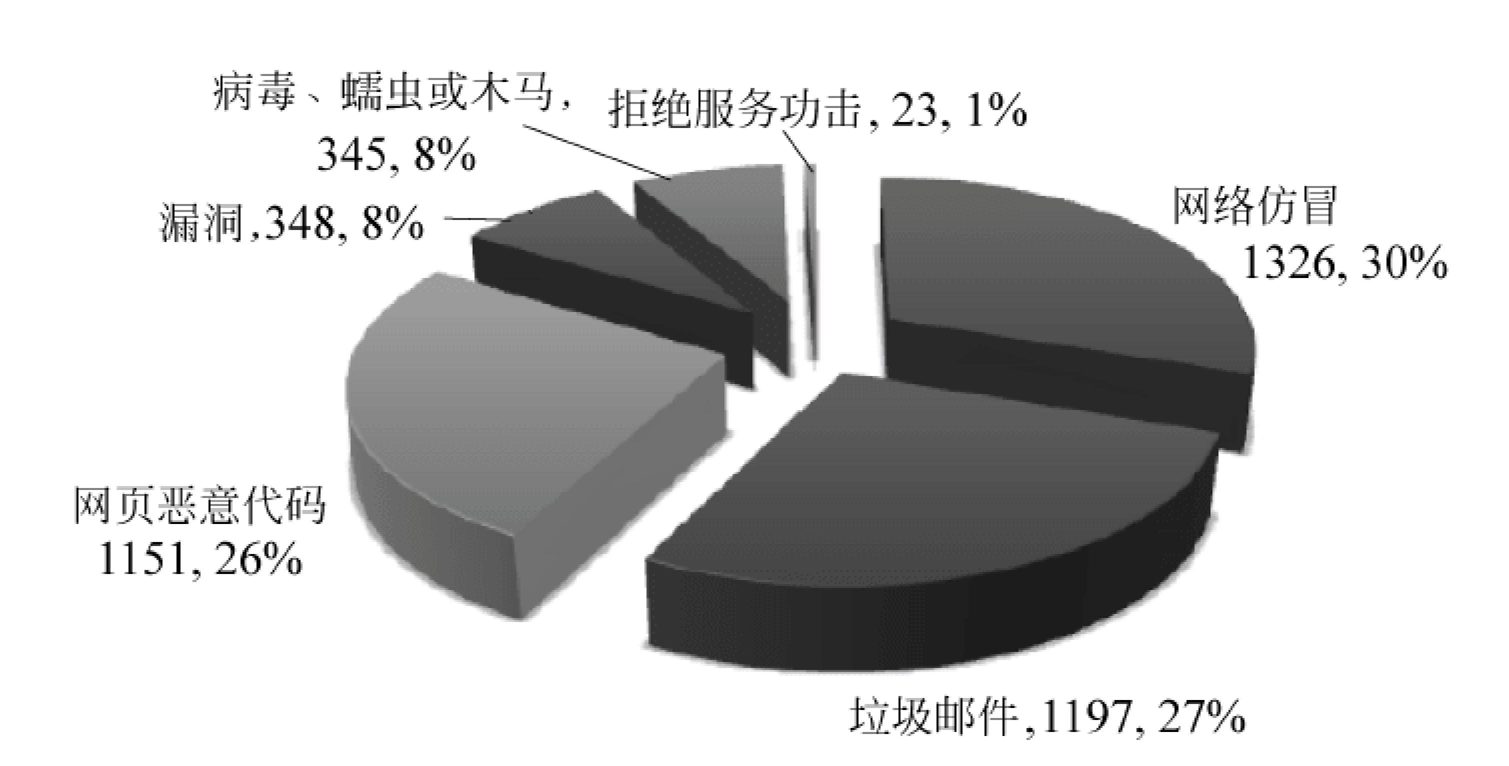


图 1.4.1　2007 年 CNCERT/CC 事件报告类型分布

与 2006 年相比,主要类型的安全事件数量均近成倍增加,网络仿冒事件数量由 563 件增加至 1326 件,增长近 1.4 倍;垃圾邮件事件数量由 587 件增加至 1197 件,增长达 1 倍;网页恶意代码事件数量由 320 件增加至 1151 件,增长近 2.6 倍。出现如此众多的攻击和破坏行为的最主要、最根本原因是网络系统存在可以被渗透的脆弱性,或称作安全漏洞。脆弱性的来源是多方面的,存在于系统设计、实现、运行和管理的各个环节。长期以来网络体系结构的研究主要考虑了如何提高数据传输的效率,构成 Internet 的一些早期网络协议也很少考虑安全问题,而且 Internet 在拓扑和新生技术等方面都是动态发展的,加之网络的开放性,使得发起攻击一般是很迅速、很容易和廉价的,并且难以检测和追踪。即便网络体系结构设计得很完美,设备软件、硬件在实现过程中的脆弱性也不可能完全避免。此外,Internet 的爆炸性发展,为了保障网络的安全运行客观上需要大批训练有素、经验丰富的网络管理工程人员,然而现实并没有得到满足,造成大量的人为操作失误,安全机制和管理政策之间的不一致性也时常出现。

尽管信息网络的安全研究已经持续多年,但对网络攻击和破坏行为的对抗效果并不理想,仍然面临着严峻的挑战。现有以防火墙、入侵检测和病毒防范等组成的网络安全系统,在功能上孤立、单一,大多只能对抗已知攻击,缺少对网络系统故障和人为操作失误等因素的处理,在体系结构上多是外在附加、被动地防御,未能解决脆弱性的本源问题,无法应对具有多样、随机、隐蔽和传播等特点的攻击和破坏行为,而且安全系统自身的安全可靠性未得到保证。另一方面,客观存在的网络攻击方式呈现出智能化、系统化、综合化趋势,新的攻击方式不断涌现,势必造成当前的安全系统规则膨胀、误报率增多、安全投入不断增加、维护与管理复杂甚至难以实施,极大地降低了信息系统的使用效率。

事实上,就整个网络安全需求而言,许多问题的研究是相关联的,分散孤立的应对方式显然不可取[55]。如何在网络复杂异构的环境下,提供一致的安全服务体系结构,如何在网络固有的脆弱性、人为的操作失误和管理漏洞以及网络攻击和破坏客观存在的状况下,保障

网络服务的可生存性,如何在保证网络高效互通的基础上,提供强大的监控能力,都是必须综合考虑的重要问题。正如美国工程院院士 David Patterson 教授所指出的,过去的研究以追求高效行为为目标,而今天计算机系统需要建立高可信的网络服务,可信性必须成为可以衡量和验证的性能[56]。构建一个安全、可生存和可控的可信网络正在成为人们关注的焦点。

解决人们对网络日益增加的依赖性与安全服务能力的有限性之间的矛盾,是进一步推进网络理论技术研究,提高网络建设及应用水平的重大问题。现实的发展对网络安全提出了更高的要求,希望在保障信息私密性、完整性和可用性的同时,能够保障网络系统的安全性、可生存性和可控性。然而,目前互联网中普遍存在的脆弱性导致了它是不可完全信任的。未来的网络安全供给模式应该是提供系统的安全服务,一方面安全应成为嵌入到网络内部的一种服务,同时要从体系结构的设计上保障网络服务的安全持续。这也是我们研究可信网络的重要目标。

尽管人们提出可信系统的概念已经有一段历史,增强计算机终端可信性的可信计算也是近年来的一个研究热点,但是国际上对可信网络的探索刚刚开始,基本概念和科学问题的认识还不够深入。目前国际上对可信性比较有代表性的阐述主要有:ISO/IEC 15408 标准中指出,一个可信的组件、操作或过程的行为在任意操作条件下是可预测的,并能很好地抵抗应用程序软件、病毒以及一定物理干扰所造成的破坏;微软公司的比尔·盖茨认为可信计算是一种可以随时获得的可靠安全的计算,并包括人类信任计算机的程度,就像使用电力系统、电话那样自由、安全[57];Algridas 和 Laprie 等人则将可信性表述为系统提供的服务可以被论证为可信任的,系统能够避免出现不能接受的频繁或严重的服务失效[58]。

一个可信的网络其行为及结果是可以预期的,能够做到行为状态可监测、行为结果可评估、异常行为可控制。具体而言,网络的可信性应该包括一组属性,从用户的角度需要保障服务的安全性和可生存性,从设计的角度则需要提供网络的可控性。不同于安全性、可生存性和可控性在传统意义上分散、孤立的概念内涵,可信网络将在网络可信的目标下融合这 3 个基本属性,围绕网络组件间信任的维护和行为控制形成一个有机整体。

如图 1.4.2 所示,信任信息的维护过程可以分为信任信息输入、信任信息处理和信任等级或策略输出这 3 个部分。信任信息采集提供具体的输入方式,主要包括:集中式安全检测,即通过在网络中设置专门的服务器,对某个范围内的网络节点进行脆弱性检测等信任信息的采集,其特点是网络结构简单,但是可扩展性相对于分布式节点的自检方式较差;分布式节点自检,即将部分监测功能交由网络节点中的代理完成,网络只负责接收检测结果,其特点是工作效率高,但是控制机制较为复杂;第三方通告,即由于不能直接对被测节点进行检测等原因,而间接地获得有关信息。

信任信息经过存储、传播和分析后,通过信任等级和策略输出用于驱动和协调需要采取的行为控制。典型的行为控制方式有:访问控制,即开放或禁止网络节点对被防护网络资源的全部或部分访问权限,从而能够对抗那些具有传播性的网络攻击;攻击预警,即向被监控对象通知其潜在的易于被攻击和破坏的脆弱性,并在网络上发布可信性评估结果,报告正在遭受破坏的节点或服务;生存行为,即在网络设施上调度服务资源,根据系统工作状态进行服务能力的自适应调整以及故障的恢复等;免疫隔离,即根据被保护对象可信性的分析结

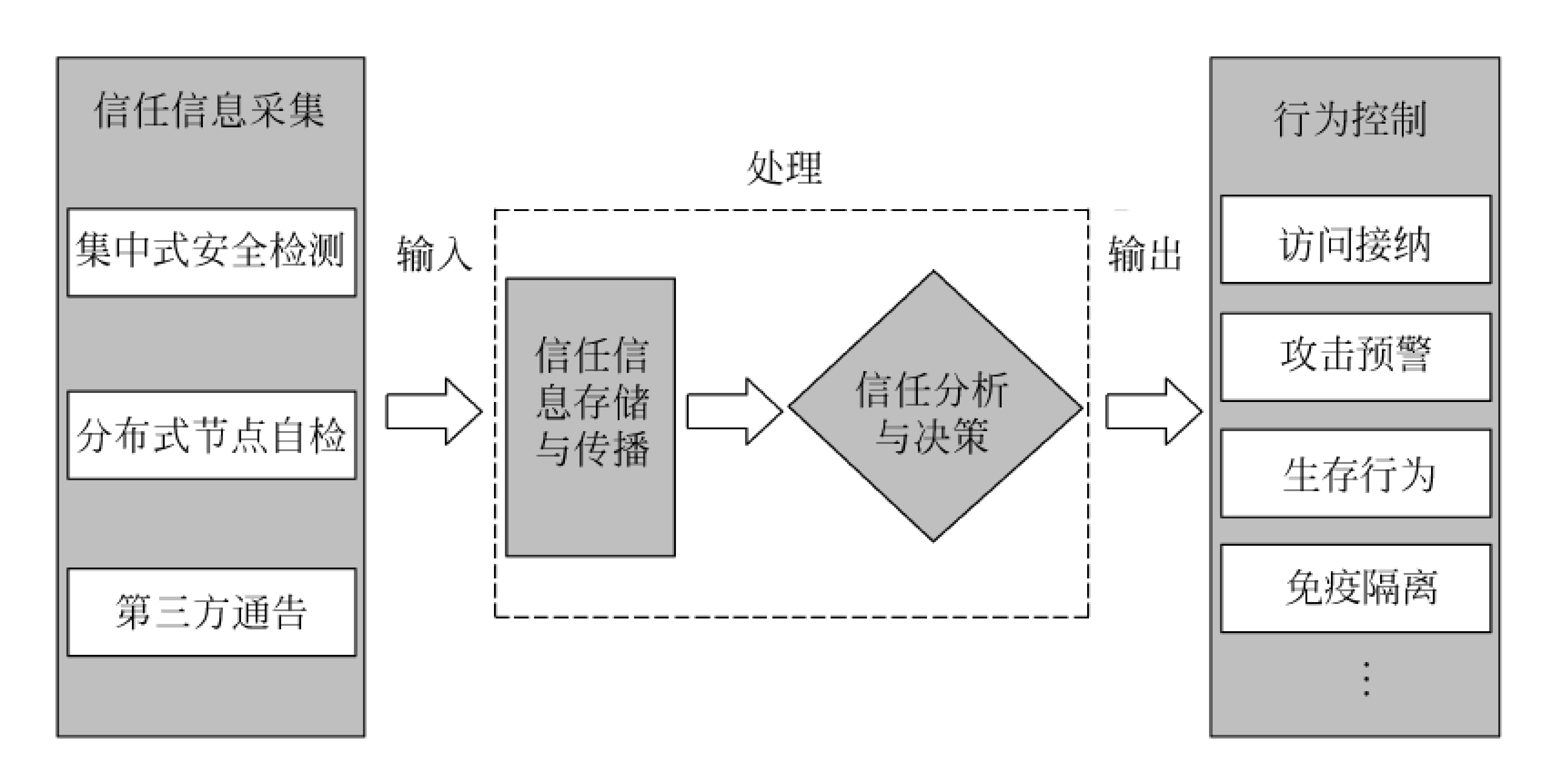


图 1.4.2　可信网络的信任维护与行为控制

果，提供到网络不同级别的接纳服务。不同于访问控制主要针对的是防护区域外具有攻击和破坏性的节点及行为，免疫隔离更多的是在攻击和破坏行为出现前主动对防护区域内的设施进行处理。

可信网络 3 个基本属性的紧密联系体现在：

（1）通过可信网络安全体系结构设计，改变传统打补丁、附加的安全供给模式，降低整个信任信息维护链体系结构设计上的脆弱性，支持多样的信任信息采集方式，保障信任信息可靠而有效地传播，并能有效地协同各种行为控制方式，使其能在可信的目标下得到融合；

（2）通过可生存性设计，在系统脆弱性不可避免以及攻击和破坏行为客观存在的状况下，提供资源调度等提高服务生存性的行为控制，提高包括安全服务在内的关键服务的持续能力；

（3）通过可控性设计，完成对网络节点的监测以及信任信息的采集，根据信任分析决策的结果实施具体的访问接纳和攻击预警等行为控制手段，从而建立起内在关联的异常行为控制体系，结束当前安全系统分散、孤立的局面，全面提升对恶意攻击和非恶意破坏行为的对抗能力。当然，还需要建立网络与用户行为的可信模型，为信任信息的分析决策、行为控制方式的选择及实施效果的评估提供判据。

1.4.1.2　网络与用户行为的可信模型

建立包括网络的脆弱性分析以及用户攻击行为描述等内容的可信模型理论，是进行可信性评估、区分网络是否被正常使用的基础，也是对抗攻击的前提。可信模型要能抽象而准确地描述系统的可信需求且不涉及具体实现细节，并可通过数学模型的分析方法找到系统在安全上的漏洞。可信模型的形式化描述、验证和利用能够提高网络系统安全的可信度。然而现时的网络已经演变成为一个庞大的非线性复杂系统，网络节点间的协议交互以及用户之间的合作与竞争，使网络行为呈现出相当的复杂性和非线性，而且攻击和破坏行为也呈现出多样、随机、隐蔽和传播等特点，从而难以预测、分析和研究。另一方面，传统理论方法具有局限性，难以建立描述网络和用户行为的可信模型。这需要借助现有的基础理论并创建新的理论，开发新的研究方法，才能逐步解决。

已有基于规则的脆弱性分析方法是从已知的案例中抽取特征，并归纳成规则表达，将目

标系统与已有的规则一一匹配。因此,规则的生成是十分关键的。对于单个的系统组件,生成规则可能并不困难,但是对于一个庞大而复杂的网络系统,就需要对大量系统组件的交互关系相当了解才可能归纳出所需要的规则。显然,这在操作上是相当困难的。此外,基于规则的方法也只能描述已知攻击方式的行为,难以应对攻击方式繁多、频繁异变的状况。基于模型的脆弱性方法为整个系统建立模型,通过模型可以获得系统所有可能的行为和状态,利用模型分析工具产生测试例,对系统整体的可信性进行评估。模型的建立比规则的抽取要简单,而且能够发现未知的攻击模式和系统脆弱性,因而适合于对系统进行整体评估。基于模型方法的关键在于模型的建立。如果模型过于简单,不能清晰描述系统可能的行为,则会导致评估结果不全面;相反,如果模型过于复杂,则可能导致评估十分困难[59]。

尽管使用模型来定量评估计算机系统的可靠性,在理论和技术上已经有了较长的发展历史,如各种组合方法、马尔可夫回报模型、离散事件仿真等,但是网络系统的安全评估大多还是采用形式化方法对整体设计的局部进行分析,缺乏定量的评估模型。如果将网络攻击和破坏行为也理解为影响系统可靠性的故障因素,则基于模型的系统可靠性的评估方法有可能用于评估网络系统可信性。但是,需要注意的是,攻击和破坏行为具有人为主观性,而一般意义上的系统故障具有很强的偶然性,因此模型评估方法急需拓展[60]。

1.4.1.3　可信网络体系结构

互联网在设计之初对安全问题考虑不足,是导致当前网络众多脆弱性的一个重要因素。一段时间以来网络体系结构的研究过度集中于如何提高数据传输的效率,形成了如今核心简单、边缘复杂的 Internet 体系模型。这种核心网络的简单性方便了新业务的部署,但同时造成核心网络对业务过于透明,基本不存在特定于应用的运行模式,导致难以检测到应用业务层面出现的问题,更难将攻击行为和新业务区分开来[61]。

此外,网络安全已经超出传统信息安全的可用性、完整性和私密性的内涵,服务的安全作为一个整体属性为用户所感知的趋势日益凸现。然而目前的许多网络安全设计很少触及体系结构的核心内容,大多是单一的防御、单一的信息安全和打补丁附加的机制,遵从“堵漏洞、做高墙、防外攻”的建设样式,以共享信息资源为中心,在外围对非法用户和越权访问进行封堵,以达到防止外部攻击的目的。在攻击方式出现复合交织的趋势下,当前的安全系统将变得越来越臃肿,严重地降低了网络性能,甚至破坏了系统设计开放性、简单性的原则。并且,安全系统自身在设计、实施和管理各个环节上也不可避免地存在着脆弱性,严重影响了其功效的发挥。因此基于这些附加的、被动防御的安全机制上的网络安全是不可信的。另一方面,网络安全研究的理念已经从被动防御转向了积极防御,需要从访问源端进行安全分析,尽可能地将不信任的访问操作控制在源端[62]。因此,可信网络的研究必须重新审视互联网的体系结构设计,减少系统脆弱性并提供系统的安全服务。尽管在开放式系统互联参考模型扩展部分增加了有关安全体系结构的描述,但只是给出了一个概念性的框架且不完善[63]。目前广泛使用的 TCP/IP 协议也缺乏完整的安全参考模型,不能在实现网络可信这一目标下,融合安全性、可生存性和可控性。

可信网络体系结构研究必须充分认识到网络的复杂异构性,从系统的角度保障安全服务的一致性。现实的互联网涵盖了不同类型的传输技术,如有线和无线,存在着不同属性的业务,如数据、图像、语音和视频。这些差异可能会形成对网络可信性威胁因素的不同关注,

然而来自用户的安全服务要求却是明确的,并不会因为某个业务需要跨越几个无线和有线的传输路径而发生改变,当然也不会关心提供安全服务的具体技术细节。作为网络研究最有价值的经验,开放系统互联应该是可信网络体系结构研究需要遵从的一个原则。

图 1.4.3 给出了可信网络的一种可能的体系结构模型。数据传输平面负责承载业务,并保障协议的可信性。可信控制平面则包括一组可信协议,提供完备、一致的控制信令,实现对用户和网络运行信息的分布式采集、传播和处理,支持信任信息在可信用户间的共享,并驱动和协调具体的行为控制方式。数据平面接受可信控制平面的监管,可信控制平面则向数据平面开放某些访问接口,从而使得业务能够获知网络运行是否可信,网络也可以根据用户要求为业务定制某种模式的运行方式,授予更高的信任级别,体现更高的可信保障水平。

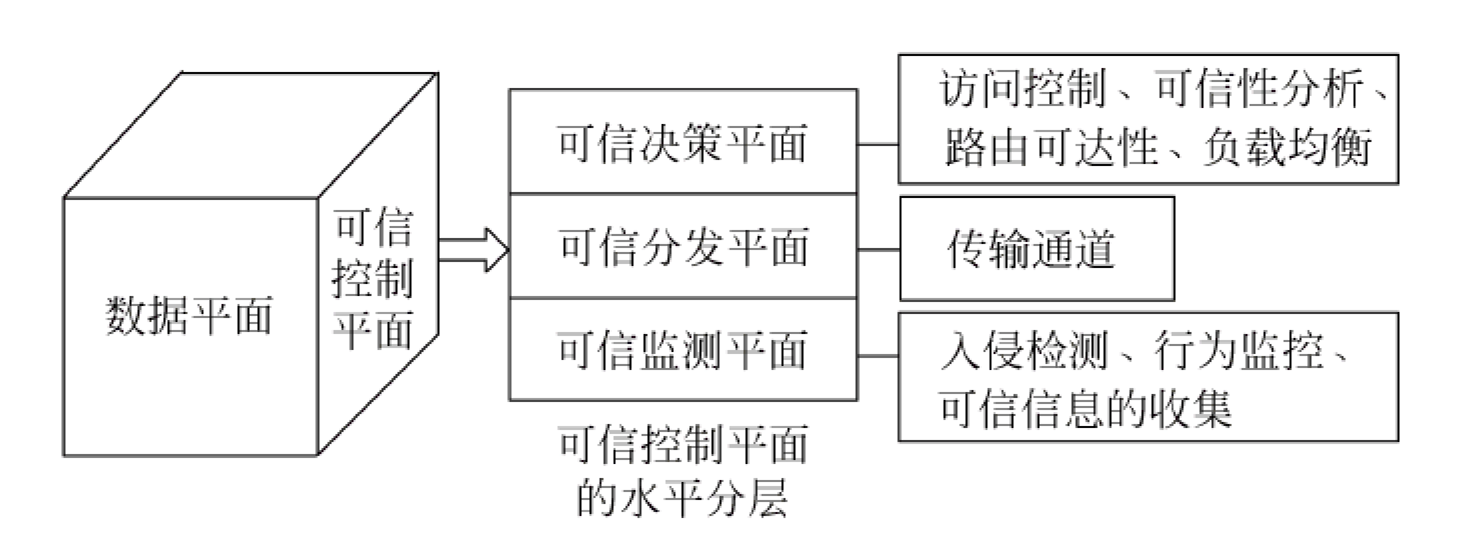


图 1.4.3 可信网络的一种体系结构模型

1.4.1.4 服务的可生存性

可生存性是网络研究的一个基本目标,是指对网络系统基本服务可用性的保障[64],即在遭受恶意攻击和发生故障时仍能按照需求及时完成任务的能力[65],或者重新配置基本服务的能力[66]。可生存性设计需要使系统能够自测试、自诊断、自修复和自组织,从而维持关键服务的关键属性,如完整性、机密性、性能等。安全服务作为网络系统的关键服务,某种程度的失效就可能会造成整个系统遭受更大范围的攻击,导致更多服务的失效甚至是系统瘫痪。由于网络系统固有的脆弱性以及人为的管理漏洞和操作失误,完全安全的网络系统是不可能存在的。因此,如何在这样一个条件下,尽可能地减少包括安全服务在内的关键服务的失效时间和失效频度,并允许网络服务的降级使用,是可信网络研究的一个关键问题。

造成网络服务失效的因素有很多,可以是系统运行过程出现的软、硬件故障,也可能是网络攻击或破坏等用户行为,甚至是一些自然因素。因此,必须在理论上深入剖析独立于具体因素的可生存性的本质特征。容错、容侵和面向恢复的计算是几种典型的提高网络服务可生存性的方式。容错主要针对网络系统内部的故障,采用故障检测、故障容许、故障纠正等技术,减少系统对外界用户表现出来的错误的状态变迁[67]。容侵则主要对抗用户的破坏和攻击行为,保障向合法用户提供服务的连续性,当然,在性能上允许一定的衰减[68]。与容错和容侵主要是避免服务失效不同,面向恢复的计算则用于解决如何在服务失效后能够快速恢复。

事实上,可生存性在某种程度上可以理解为对冗余资源的调度问题。图 1.4.4 给出了一种可生存系统的体系结构设计[69]。代理和服务器在功能上是冗余的,并且各冗余组件在

设计方案和实现技术上尽可能地不相关。冗余代理按照某种规则推选或改选出一个主代理，负责在多个服务器之间调度客户端的任务请求，并检查各服务器的工作状态，将失效服务器上正在进行的工作重新分配给另一个服务器，从而对容侵系统的外部用户保持服务的连续性。当然，这里存在性能评价和优化的问题，例如如何用最小的资源成本获得最大的可生存性能，以及针对不同的资源冗余方式设计最优的任务调度算法。尤其是为了实现服务的可恢复，更需要关注因服务窗口失效而产生的任务再调度问题，显然，这些任务一般要求比正常进入系统的任务具有较高的处理级别。

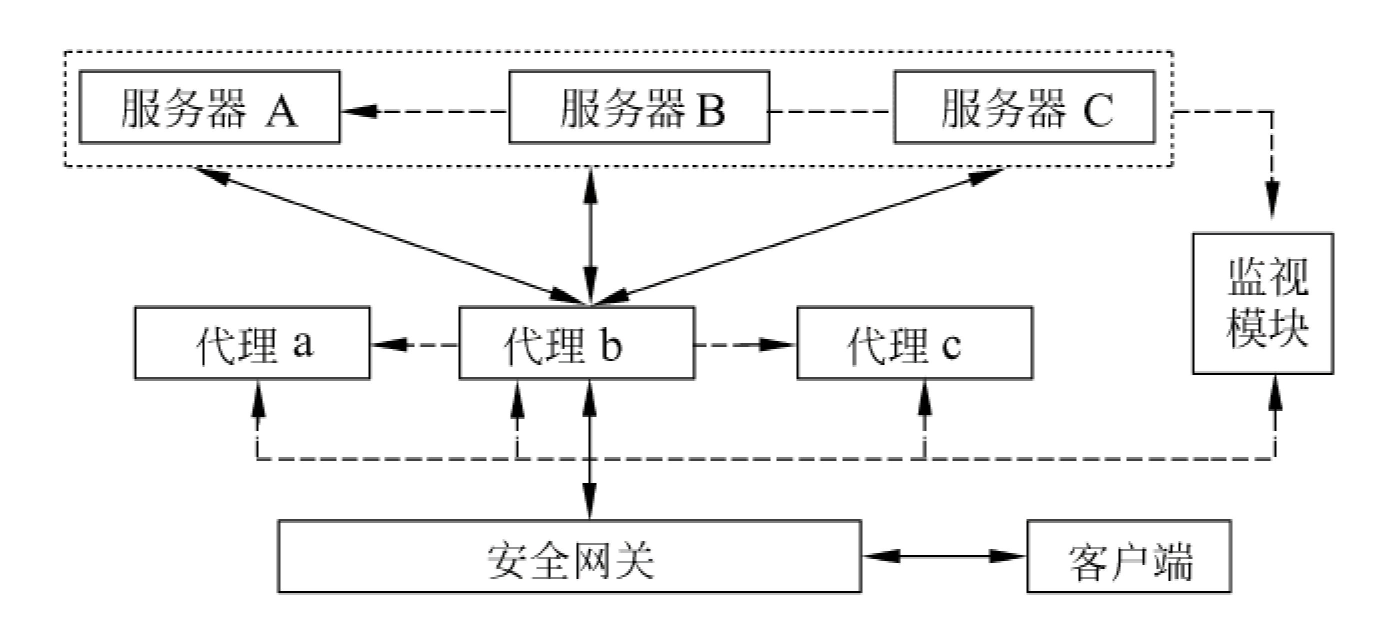


图 1.4.4　一种可生存系统的体系结构设计

1.4.1.5　网络的可控性

互联网络发展至今，已成为一个庞大的非线性复杂系统，如系统规模和用户数量巨大且不断增长、协议体系庞杂、业务种类繁多、异质网络融合发展等。这远远超出了当初设计的考虑，现有的一些控制手段相对而言显得很薄弱，产生了许多安全隐患。“边缘论”和面向非连接的设计思想保障了网络的高效互通，逐跳存储转发的分组传送方式简单、灵活，无需在中间节点维护过多的状态信息，核心网络的工作集中于路由转发。这些机制的优点是设计简单、可扩展性强等，然而却造成了分组传输路径的不可控，网络中间节点对传输数据包的来源不验证、不审计，导致地址假冒、垃圾信息泛滥，大量的入侵和攻击行为无法跟踪[70]。

如何解决网络的低可控性与安全可信需求之间的矛盾，建立内在的、关联的网络可控模型，在理论和技术上仍是当前学术界的一个难题。网络的可控性是可信网络在设计上的一个重要属性，主要目标是：在网络的关键部分增加认证、授权等控制机制使网络更可信；在网络中维护一定的状态信息，施加必要的控制，使网络具有某种程度面向连接的特性；在不同的层次上实施对网络的监管，提供采集和传输网络组件信任信息以及检测网络运行状态的机制，并提供异常行为控制和攻击预警的快速算法。

此外，网络攻击和破坏行为的综合化，客观上要求对抗机制也要综合化，然而当前的网络安全系统是分散而孤立的，如入侵检测不能对抗蠕虫病毒，防病毒软件不能对抗拒绝服务攻击，防火墙对病毒攻击和木马攻击也无能为力。因此可信网络的可控性设计必须能够建立内在关联的监控体系，完成对网络节点的监测以及信任信息的采集，并根据信任分析决策的结果实施具体的访问接纳和攻击预警等行为控制手段，使得多样的监控机制能够融合在一个可信的平台下发挥效用。

1.4.1.6 国际相关研究工作

鉴于当前网络安全系统分散、孤立的现状以及用户对系统的安全服务的迫切需求，国际上都在积极探索新的研究思路。尽管可信系统这一概念的提出已经有一段时间，针对计算机系统的可靠计算或容错计算也有了较为深入的工作，但是网络系统的可信性问题还是近年来随着人们对网络安全的日益重视才提出的。而且，当前的工作大多是就可信网络在理论与技术的某个局部目标展开的，并没有形成完整的体系。可信网络的许多概念还处在摸索阶段，尤其是对其基本属性和面临的关键问题并没有清晰而一致的描述。

在可信终端的研究方面，为了解决信息终端结构上的不安全性，从基础层面上提高其可信性，国际上正在推动可信计算技术[71~73]。1999 年，由康柏、惠普、IBM、Intel 和微软牵头组织了可信计算平台联盟(Trusted Computing Platform Alliance，TCPA)，致力于在计算平台体系结构上增强其安全性，为高可信计算(trustworthy computing)制定开放的标准。2003 年 TCPA 改组为可信计算组(Trusted Computing Group，TCG)，成员扩大到 200 家，并发布了可信平台模块(Trusted Platform Module，TPM)规范。国际上一些著名公司也在积极研发支持高可信计算的产品。如微软公司的 Palladium 计划(后改名为 NGSCB)，准备设计支持高可信计算的新版 Windows 操作系统；此外，还有 Intel 公司的 LaGrande 技术以及 IBM 嵌入式安全系统等。

在容错研究方面，也在向提高系统可信性的方向演化。容错性是可靠性的一个重要内容，其概念最初出现在 1830 年英国数学家和分析仪发明者设计的巴比奇计算工具中。1970 年，IEEE-CS 容错计算技术委员会成立，着重讨论计算机可靠性问题；1971 年起，IEEE 的容错计算委员会主持召开了首次 FTCS 大会，研究讨论计算机系统的可信赖技术及其进展。之后，IEEE CS 建立了 IFIP WG 10.4 "Dependable Computing and Fault Tolerance"小组。1992 年，IFIP WG 10.4 的成员在 J. C. Lapire 的带领下出版了专著"Dependability: Basic Concepts and Terminology"。1999 年，IEEE 泛太平洋容错系统会议改名为"可信计算会议"。2000 年，"IEEE 国际容错计算会议(FTCS)"与国际信息处理联合会(IFIP)的 10.4 工作组主持的"关键应用可信计算工作会议"合并，改名为"IEEE 可信系统与网络国际会议(ICDSN)"。

网络安全监测的研究始于 20 世纪 90 年代，比较著名的有美国国家安全实验室研究的通用入侵监测架构，以及美国航天署提出的提高系统安全性的安全监测架构等，初步形成了各不相同的安全监测技术体系。此外，现阶段国际上在安全政策领域的研究工作刚刚起步。目前关于安全政策的主要研究组织是 IETF 的 Policy Framework 工作组，正致力于建立一种通用政策架构，在大型网络环境中管理和分配政策，达到政策的一致实现。目前该工作组已在政策架构定义语言、通用开放政策服务等方面产生了一批协议草案。

基于不同的研究背景，国际上也有些工作正试图描述可信网络，一些机构也启动了相应的研究计划。例如，DARPA 的 CHAT(Composable High-Assurance Trustworthy Systems)项目，就探讨了如何在对安全性、可靠性、可生存性及其他必要属性具有严格要求的条件下，得到可以验证的可信系统和网络[74]。卡内基·梅隆大学也在推动 TRIAD (Trustworthy Refinement through Intrusion-Aware Design)项目，研究如何借助入侵检测提高网络系统的可信性[75]。加州大学伯克利分校与斯坦福大学合作研究面向恢复的计算

(Recovery-Oriented Computing, ROC),强调了服务失效后的恢复[56]。杜克大学的 Yumerefendi 和 Chase 将可审计性(accountability)作为可信网络系统设计的核心目标,认为可审计系统的行为、状态和动作应该是不可否认的(undeniable)、可确认的(certifiable)和防篡改的(tamper-evident)[76]。哈佛大学的 Camp 则提出,下一代 Internet 可信系统必须建立在从社会科学中获得的人类信任的概念之上,可信互联网络将具有私密、安全和可靠等多个尺度,必须将它们集中到信任这个统一目标上来才能有效保障网络的安全可信[77]。麻省理工学院的 David Clark 则为 Internet 提出了知识平面的概念,用于采集和汇聚网络运行的信息,提高网络的可控性。

此外,由于媒介的开放性、带宽和计算资源的有限性、数据传输的不可靠性、拓扑位置的频繁变化、在许多场合下难以实施集中控制和管理等特点,与固定网络相比,无线移动环境下的网络可信性问题面临着更严重的威胁,现实的需求更为迫切[78]。相关的代表性工作主要有:可信传感器网络[79]、Ad hoc 网络的信任评估[80]和移动 IPv6 的信任管理[81]等。

我国在容错、可信计算、安全监测等领域也进行了多年的研究,部分工作达到了国际先进水平,但在可信网络方面还缺乏整体认识,较多地处于跟踪国外研究动态的阶段。国内从 20 世纪 80 年代中期开始研究计算机系统的容错。这方面的工作主要是由中国计算机学会容错专业委员会推动,并有了不错的进展。2003 年,TCG 经由联想集团和微软公司第一次被介绍到国内,并于 2004 年在北京召开了"高可信计算标准研讨会"。2004 年在我国还相继召开了"中国首届可信计算平台技术论坛"和"第 1 届中国可信计算与信息安全学术会议",很好地促进了可信计算机终端的研究。

1.4.2 可信网络访问控制

1.4.2.1 可信和信誉的概念

很多文献中都提到了可信(trust)的概念,图 1.4.5 为文献[82]中提出的可信的分类。从图中可以看到可信行为产生的过程,即可信行为在产生之前,信任受到意图的影响,并表现为可信决策。可信是协同环境的基础,是一种连续的行为而不是离散的,通常不具有继承性,但是通过学习过程可以对可信值进行动态更新。一般从两个方面来描述可信,即时间(time)和上下文(context):时间表现了可信的动态性,一个不可信的用户可能通过良好的行为得到一个较高的可信值,而一个可信用户也可能故意表现为不诚实的行为;可信是上下

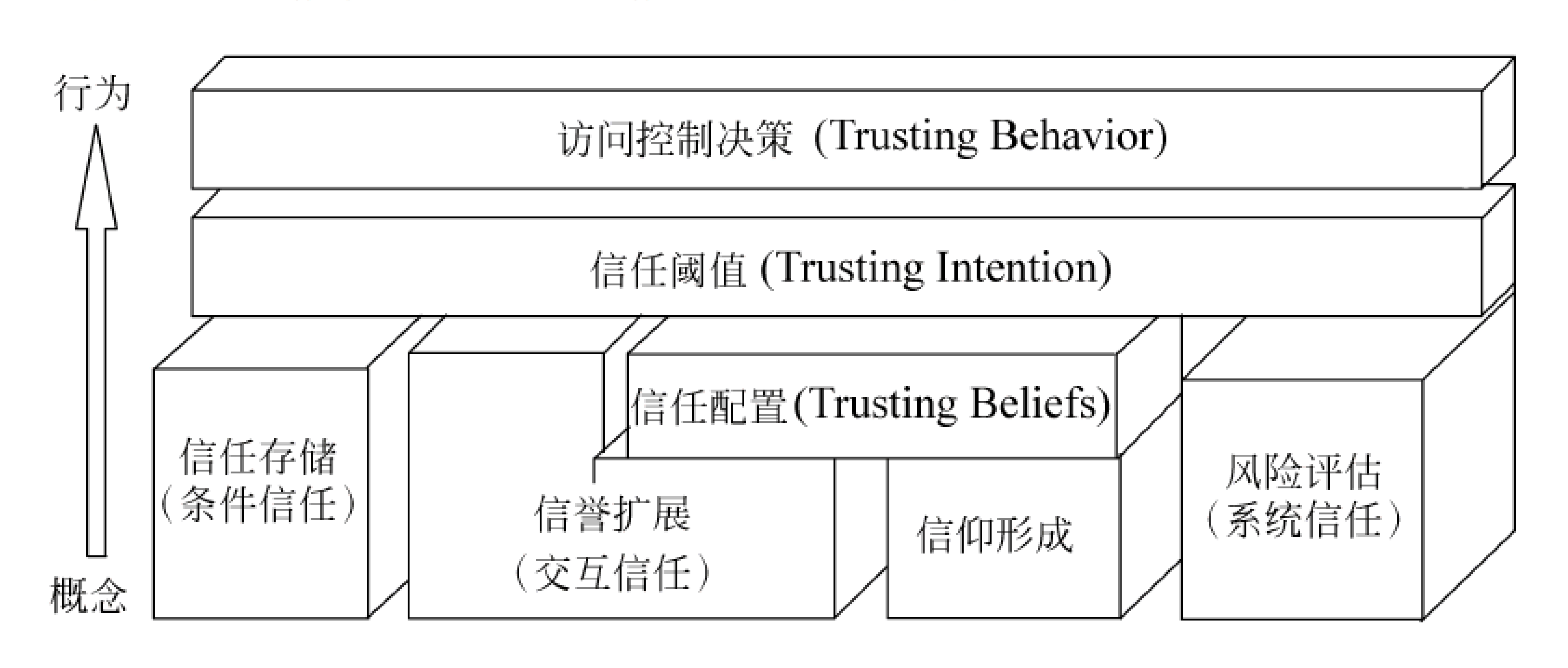


图 1.4.5 可信类型和结构[82]

文相关的，在不同方面得到的可信值是不同的，如用户 A 对用户 B 某一方面的行为是可信的，但是用户 A 不一定对用户 B 其他方面的行为可信。无论是有线网还是无线网，对可信网络访问控制的研究要求提出一种基于可信的访问控制机制，其可信模型都有一些基本的特点，如：可信是一个主观的概念；可信值可以是正值也可以是负值；一般可信值取在[0,1]区间，且是建立在历史经验上的一个连续值；可信信息在节点间可以相互交换，并被动态更新。

可信通常与信誉（reputation）相混淆，这里有必要将这两个概念加以区别[83]：可信是主动的，它是一个用户对另一个用户某种能力的信任，建立在以往交易的满意度评估上；信誉是被动的，是其他用户通过观察和交互过程对该用户的评价，一个用户的信誉值在不同情况和行为下都是不同的，且信誉是可信信息的集合，它是通过交互或资源共享的历史行为来预测该用户行为是否可信。虽然可信和信誉是两个不同的概念，但却都是上下文相关的，都具有多样性和动态性。

由于 Ad hoc 网络主要应用在军事系统中，对可信和信誉的要求程度较高，因此对信誉系统的研究较多。目前信誉系统有 3 类：正信誉（positive reputation）、负信誉（negative reputation）和两者的结合。正信誉系统仅考虑节点正面的行为和反馈，如在 CORE[84] 系统中节点之间通过协作建立正信誉值，其缺点是仅考虑了节点正面的行为信息，而没有考虑负面的反馈。负信誉系统[85]仅考虑节点负面的行为和反馈，其前提是假设系统中节点是可信的，通过其行为反馈来更改节点的信誉值，其缺点是缺少惩罚机制，对那些信誉值较低且表现为恶意行为的节点无能为力。CONFIDANT[86]系统同时考虑了正信誉和负信誉，并通过等级表（rating list）和权重（weighting）来计算节点的信誉值，使具有恶意行为的节点在网络中孤立，激励所有节点参与到网络中，提高了系统的通信效率。

近年来，对网络可信模型的研究已经引起关注，多数都是基于可信和信誉的模型，这些模型都可以应用到网络的可信访问控制研究中。文献[87,88]是为 P2P 系统提出的基于贝叶斯网络（Bayesian network）的可信和信誉模型，其信誉建立在推荐（recommendation）的基础上。由于在 P2P 系统中对等点的可信值是多方面的，即在不同的情况下，对 peer 的可信值是不同的，贝叶斯网络为不同情况下的可信值提供了一种灵活的模型，可应用于 P2P 文件共享系统或其他 P2P 系统中。文献[89]为动态协同网提出了一个分布式基于可信的访问控制系统，该系统是一种以节点为中心的基于可信的访问控制，结合了信誉和风险（risk），具有动态性，能够激励节点之间的相互协作和共享资源，适用于移动 Ad hoc 协同环境。

1.4.2.2 可信和信誉机制

对可信网络访问控制的研究有助于提高网络系统的安全性，是可信网络安全性研究中的一个重要问题，必须将传统的访问控制方法与认证、授权、密钥管理等方法相结合才能解决可信网络的安全问题。基于角色的访问控制本身具有很多优势，如权限与角色相关联，根据责任和资格授予用户相应的角色，从而实现用户和访问权限的逻辑分离，并且管理员可以根据系统和应用程序的需要添加和撤销用户的角色或角色的权限，极大地简化了权限管理。但在可信网络中，仅通过基于角色的访问控制并不能实现网络的可信，必须通过适当的基于可信的访问控制策略，经过安全认证与授权后的用户才具有某种角色，可行使某种权限或执

行某些安全操作,这样才能保证网络中用户的行为是可信的。

在可信网络中,一个节点对另一个节点能力的信任,是建立在直接经验的基础上的;信誉是一个节点对另一个节点能力、诚实度和可靠性的信任,建立在其他点的推荐上,推荐值也称为参考值(reference)。可信网络中的节点通过询问或节点之间的交互得到一些参考值,根据这些参考值动态更新对其他节点的可信值。可信分为两类:一类是对文件提供者在提供文件共享能力上的可信;另一类是对那些提供推荐的节点可靠性上的可信。信誉值可以集中计算,如由可信第三方计算,也可以分散计算,即每个点通过询问推荐并各自计算出其他点的信誉值。

图 1.4.6 为可信网络的可信和信誉机制,从图中可以看到可信和信誉机制的实现过程:当一个节点需要选择一个可信点进行交易时,如果历史上曾有过与该点交互的经验,则在自己的可信点数据库中找到一个可信值最高的文件提供者与之交互;如果以前没有过与该点交互的经验或对该点了解较少,则从其他节点的推荐中,通过综合计算选择一个信誉值高的节点进行交互。通过这次交互的满意度对该点进行评估,并更新该节点的可信值,同时更新那些提供推荐的节点的可信值。可信和信誉机制可以帮助区分好坏节点,并找到适合自己需求的交易节点,提高了节点之间通信的效率。这种机制可以根据不同的需求应用于不同网络环境的基于可信的访问控制模型中,以提高网络节点间通信的效率和安全性。

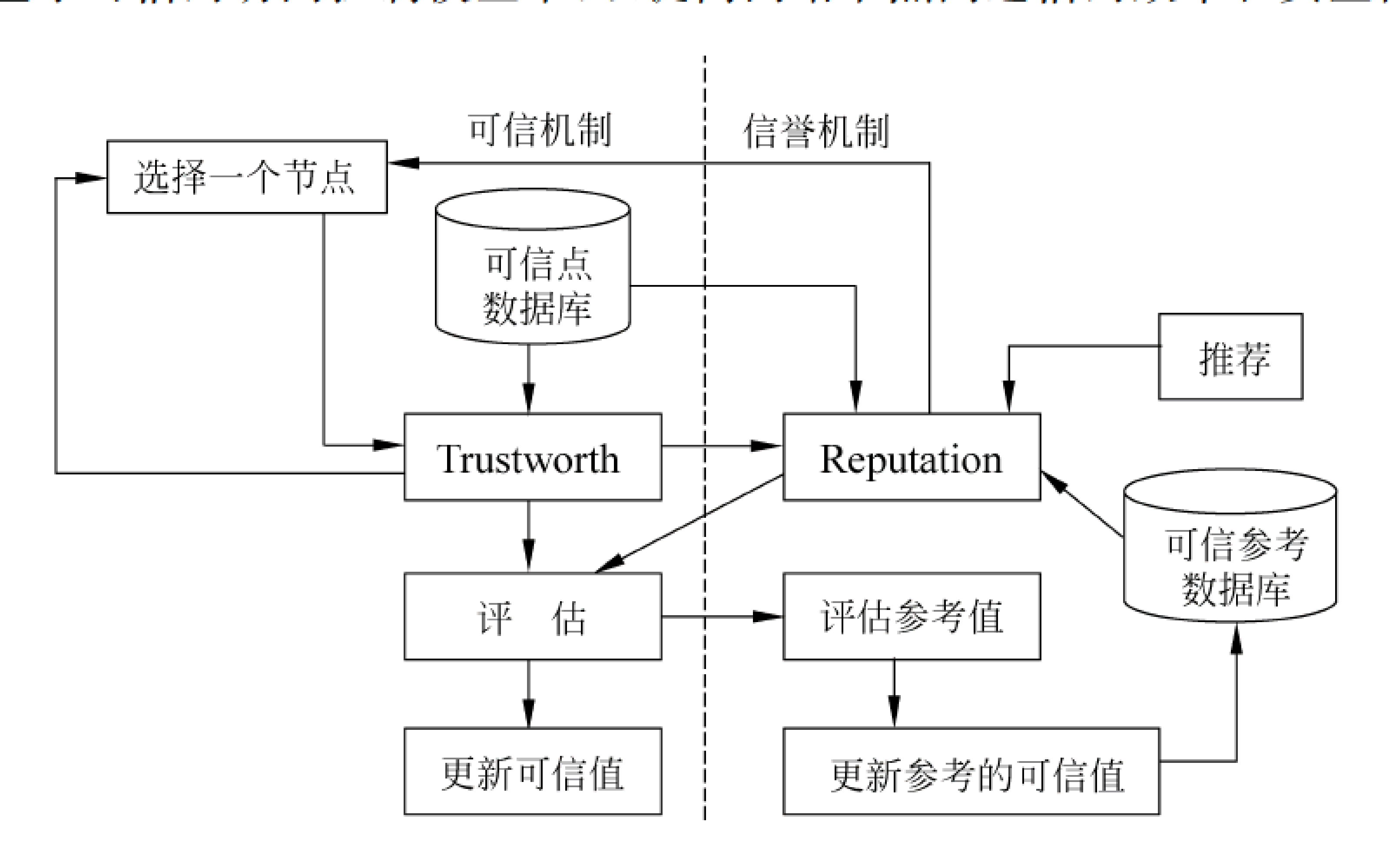


图 1.4.6　可信网络中可信和信誉机制

可信网络的访问控制要求建立基于可信的访问控制模型。由上面的介绍可以看到,可信访问控制主要是建立基于可信和信誉的模型。目前研究较多的是针对 P2P 文件共享系统和 Ad hoc 网络的可信模型,由于这两种网络都要求节点间共享资源,对可信的要求程度较高。可信值的评估方法在文献[90,91]中都有介绍,而针对 P2P 系统中可信评估的研究也越来越多[92~94]。这些可信模型和可信评估的方法都将为可信访问控制提供参考,并成为可信网络研究的一个重要内容。

可信网络作为可信计算发展的必然趋势,对可信访问控制的研究也必然成为热点。除了建立可信模型和评估可信值这两个重要的问题以外,还有一些关键问题需要解决,如节点的搜索问题,即如何找到可信的节点并向它们询问推荐;如何保证系统免受各种攻击等[95]。

1.4.3 可信计算

可信计算(trusted computing,TC)概念的提出是为了保护重要信息系统的敏感数据,解决日益严重的计算机安全问题。由 Compaq 牵头,HP,IBM,Intel,Microsoft 在 1999 年 10 月共同发起成立了可信计算平台联盟(Trusted Computing Platform Alliance,TCPA)[96]。TCPA 的目的是为可信计算制定开放的标准。TCPA 通过硬件和软件的标准化来防止软件攻击以及身份盗用,保证数据的安全性。TCPA 要求当前的计算平台加入硬件和软件的扩展来支持可信计算,如反篡改可信平台模块(TPM)和安全操作系统核心。TCPA 的主要特色是安全启动、平台验证、受保护的存储。

2003 年 4 月 8 日,TCPA 中的 AMD,HP,IBM,Intel 和 Microsoft 对外宣布,将 TCPA 重新改组,更名为可信计算工作组(Trusted Computing Group,TCG),并继续使用 TCPA 制定的"Trusted Computing Platform Specifications",同时也在制定符合 Palladium 的 TPM 1.2 技术规范。目前 TCG 的成员数目超过 200。TCG 的任务是制定产品的技术规范,以使用户可以保护关键数据和信息。TCG 的文档是为了在计算平台中赋予可信性的工业技术规范。该技术规范定义了可信的子系统,它是构成可信计算平台的必需部分,为操作系统和应用提供功能调用。

计算平台(platform)指的是用户可以用来运行应用程序的通用产品。计算平台包括服务器、PC、笔记本和手机等。应用程序包括操作系统等。

可信计算平台(trusted platform)是一个产生软件进程信任基础的平台,即为本地用户和远程实体信任的,其行为运作在针对某一特定目的的可期望的模式下。

可信计算平台应用研究的基本目标就是要建立一个网络中的可信任域,并基于该网络信任域的管理系统将个体的可信计算平台扩展到网络中,形成网络里的可信任域,并结合信息保密网的应用给出可信计算平台的应用方案。网络信任包括终端可信、局域网可信和网络互联可信。网络信任管理的内容包括:统一用户管理、统一资源管理、统一授权管理、证书管理、策略管理和审计管理。网络信任管理系统及其应用可达到如下目标:

(1) 避免非法用户使用本地终端及网络资源;

(2) 防止网络环境下机密信息泄露;

(3) 防止被非法用户窃取个人私密信息;

(4) 防止网络非法接入;

(5) 避免信息系统被恶意代码或木马感染,造成系统瘫痪等。

可信计算平台基本特征主要包括以下 3 点:

(1) 受保护的能力

受保护的能力指的是一组排他性访问受保护的区域的命令。受保护的区域是可安全操作敏感数据的区域(如存储器和寄存器等),这些区域的数据只能被这种受保护的能力所访问。TPM 实现了用来保护和报告完整性测量的受保护能力和受保护区域,称为平台配置寄存器 PCR。TPM 保存加密密钥用来认证报告的测量。TPM 的受保护能力包括额外的安全功能,如加密密钥管理、随机数发生器等。

（2）验证

验证(attestation)是证明信息正确性的过程。外部实体可确认受保护区域、受保护能力和信任根(root of trust)。一个平台可以验证该平台的描述特征,这些特征影响平台的完整性。验证可被理解成几个方面：TPM 的验证、平台的验证和对平台的认证。

TPM 执行的验证是一种为 TPM 可知的数据提供证明的操作。该操作通过使用 AIK 来对内部特殊的 TPM 数据进行数字签名。完整性和 AIK 本身的有效性和可接纳性由检验者来确定。AIK 可从保密的认证中心或者通过可信证实协议来获得。

平台的验证是一种操作,为该平台的一组完整性测量提供证明,通过对 TPM 中的 PCR 集合使用 AIK 数字签名来实现。平台的认证是为声明的平台身份提供依据,通过使用非移植的签名密钥实现。

（3）完整性的测量、存储和报告

完整性的测量是一个获得平台特性的度量过程,这些特性影响了一个平台的完整性。完整性存储保存这些度量,并把这些度量的数字签名放到 PCR 中。

测量的起点是测量信任根(root of trust for measurement)。一个静态的测量信任根从启动状态(如加电自检)开始测量。一个动态的测量信任根从不可信的状态转移到可信的状态。

在完整性测量和完整性报告之间的中间步骤是完整性存储。完整性存储保存完整性测量到日志中,存储这些测量的数字签名到 PCR 中。完整性报告是验证完整性存储的过程。

完整性测量、存储和报告的一个原则是平台可以允许进入任何可能的状态,包括不符合要求的状态或者是不安全的状态,但是平台不能允许谎报其过去所处的状态。一个独立的过程用来验证完整性的状态和确定相应的响应。

1.4.4 可信网络连接

当可信计算模块在单个主机逐渐广泛使用之后，解决了单机的可信问题，而网络连接的可信成为急需解决的问题。在 2004 年 5 月，TCG 成立了可信网络连接（Trusted Network Connect，TNC）分组(TNC SubGroup,TNC-SG);作为 TCG 中基础设施工作组(Infrastructure Work Group）的一部分，它将 TCG 的视野延展到了网络的安全性和完整性，设计防止不安全设备接入和破坏网络的机制。TNC-SG 的主要工作是开发定义 TNC 架构的一系列标准,TCG 在 2005 年发布了可信网络连接规范[97]。

1.4.4.1 TNC 架构

TNC 是建立在基于主机的可信计算技术之上的,其主要目的在于通过使用可信主机提供的终端技术,实现网络访问控制的协同工作,又因为完整性校验被终端作为安全状态的证明技术,所以用 TNC 的权限控制策略可以估算目标网络的终端适应度。TNC 网络构架会结合已存在的网络访问控制策略(例如 802.1x,IKE,Radius 协议)来实现访问控制功能。

TCG 设计 TNC 架构的原则是把 TNC 设计为一个开放的通用架构，这样 TNC 才能与现有大量不同网络技术和网络设备协同工作。TNC 的一个重要目标是，使用 TPM 的授权机制作为实现可信网络连接的重要组成部分,并通过提供一个由多种协议规范组成的框架

来实现一套多元的网络标准。在 TNC 架构的基础上，TCG 可以开发不同的协议，在不同的网络标准下达到这样的目标：

(1) 平台认证：确认请求访问网络的端点(或主机)的身份以及平台的完整性证明。

(2) 端点完整性认证：建立端点状态的“信赖”级别，以确保被代管应用程序的安全状态和升级情况；修改数字签名库来实现反病毒和入侵检测，防止系统被有害软件攻击。

(3) 访问策略：保证端点及其用户的身份辨别，在连接到网络之前生成端点的可信级别，这包含了许多现有或即将出现的标准。

(4) 评估(assessment)、孤立(isolation)和矫正(remediation)：确保不能达到安全策略标准的端点被孤立在网络之外。另外，如果存在合适的矫正方案(如更新软件或病毒的特征库)，则应用校正方案，以使端点能够访问网络。

TNC 架构由实体、层、组件和组件间的接口组成，如图 1.4.7 所示。TNC 架构框图中有 3 列，对应 3 个实体 AR，PEP 和 PDP。图中矩形框表示实体中的组件，3 行分别对应 TNC 架构中 3 个抽象层。

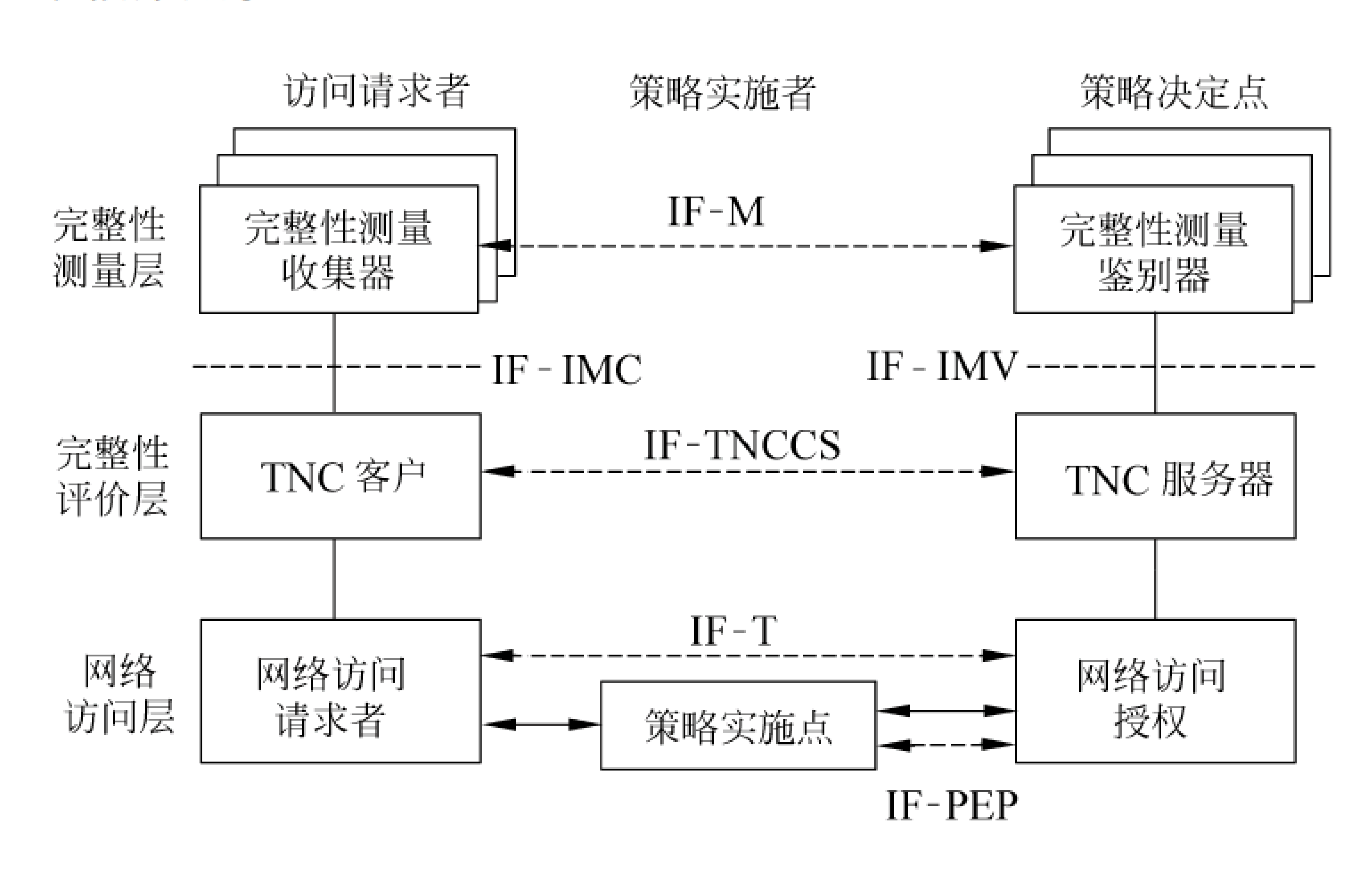


图 1.4.7 TNC 架构

1. 实体

实体(entity)是网络中具有对应角色的逻辑实体(不一定是物理实体)，TNC 架构中有 3 个实体：

访问请求者(access requestor，AR)：请求访问受保护网络的逻辑实体(可能是一台或多台物理计算机，或一个独立的程序)。

策略决定点(policy decision point，PDP)：根据特定的网络访问策略检查 AR 的访问认证，决定是否授权访问的网络实体。

策略实施点(policy enforcement point，PEP)：执行 PDP 的访问授权决策的网络实体。

当访问请求者向策略实施点发出网络连接请求时，该连接请求将首先由策略决定点进行判断，依据系统的访问策略，并根据访问请求者当前的完整性及其他安全属性，对连接请求做出判断。然后将该判断发送给策略实施点，由策略实施点来具体实施。

2. 层

根据组件的功能把不同实体中的组件分为 3 个抽象层(layer)：

网络访问层(network access layer)：包含组件的功能属于传统网络连接和安全，如 VPN，802.1x 等。

完整性评价层(integrity evaluation layer)：从不同访问策略的角度来评价 AR 实体的整体完整性。

完整性度量层(integrity measurement layer)：包含插件(plug-in)组件，这些组件的功能是为安全程序收集和校验 AR 实体中与完整性有关的信息。

3. 组件

组件(component)是实体中完成具体功能的逻辑功能模块。AR 实体中的组件有：

网络访问请求者(network access requestor，NAR)：NAR 组件的功能是发起网络请求，一个 AR 实体中可能有多个 NAR。

TNC 客户(TNC client，TNCC)：聚集了 IMC 的完整性度量信息，同时协助完成完整性检查握手(integrity check handshake)，度量并报告平台以及 IMC 的完整性。

完整性测量收集器(integrity measurement collector，IMC)：从不同安全角度度量 AR 实体的完整性。收集的信息包括操作系统安全性、反病毒软件、防火墙、软件版本等。

PEP 实体中只有策略实施点(PEP)组件，其功能是控制对受保护网络的访问，PEP 向 PDP 咨询是否授权访问。

PDP 实体中的组件有：

网络访问授权(network access authority，NAA)：决定一个 AR 是否应该得到访问授权。向 TNC server 询问 AR 的完整性是否满足 NAA 的安全策略。

TNC 服务器(TNC server，TNCS)：管理 IMV 和 IMC 之间的消息流向，收集 IMV 的行为推荐(action recommendation)并组合为 TNCS 的整体行为推荐，发送给 NAA。

完整性测量鉴别器(integrity measurement verifier，IMV)：根据从 IMC 和其他数据得到的度量校验 AR 的完整性。

4. 接口

接口(interface)定义了组件之间的协议和消息。

完整性度量收集接口 IF-IMC (integrity measurement collector interface)：IF-IMV 是 IMC 同 TNCC 之间的接口。该接口的主要功能是从 IMC 收集完整性测量值，并支持 IMC 与 IMV 之间的信息流动。

完整性度量校验接口 IF-IMV (integrity measurement verifier interface)：IF-IMV 是 IMV 与 TNCS 之间的接口。该接口的主要功能是将从 IMC 得到的完整性测量值传递给 IMV，支持 IMC 与 IMV 之间的信息流动，将 IMV 所做出的访问决定传递给 TNCS。

TNC 客户-服务接口 IF-TNCCS (TNC client-server interface)：IF-TNCCS 是 TNCC 和 TNCS 之间的接口。该接口定义了一个协议，该协议传递如下的信息：①从 IMC 到 IMV 的信息(如完整性测量值)；②从 IMV 到 IMC 的信息(如要求额外的完整性测量值)；③会话管理信息和一些同步信息。

厂商定制的 IMC-IMV 消息接口 IF-M (vendor-specific IMC-IMV messages)：IF-M 是 IMC 和 IMV 之间的接口。在该接口上传输的信息主要是一些与提供商相关的信息。

网络授权传输协议(network authorization transport protocol)IF-T：IF-T 维护在 AR 实体和 PDP 实体之间的信息传输。在这两个实体中维护该接口的组件为 NAR 和 NAA。

平台可信服务接口 IF-PTS (platform trust services interface)：提供平台可信服务，确保 TNC 组件是可信的。

策略执行点接口 IF-PEP (policy enforcement point interface)：IF-PEP 为 PDP 和 PEP 之间的接口。该接口维护 PDP 和 PEP 之间的信息传输。通过它，PDP 可以指示 PEP 对 AR 进行某种程度的隔离，以对 AR 进行修复。当修复完成之后，方可授予 AR 访问网络的权利。

1.4.4.2 TNC 信息流动

通过接口，各种信息在各个组件之间流动，图 1.4.8 为 TNC 结构中各个组件之间的信息流动示意图。这里假定认证和授权的顺序是按照用户认证、平台认证和完整性检查来进行的。

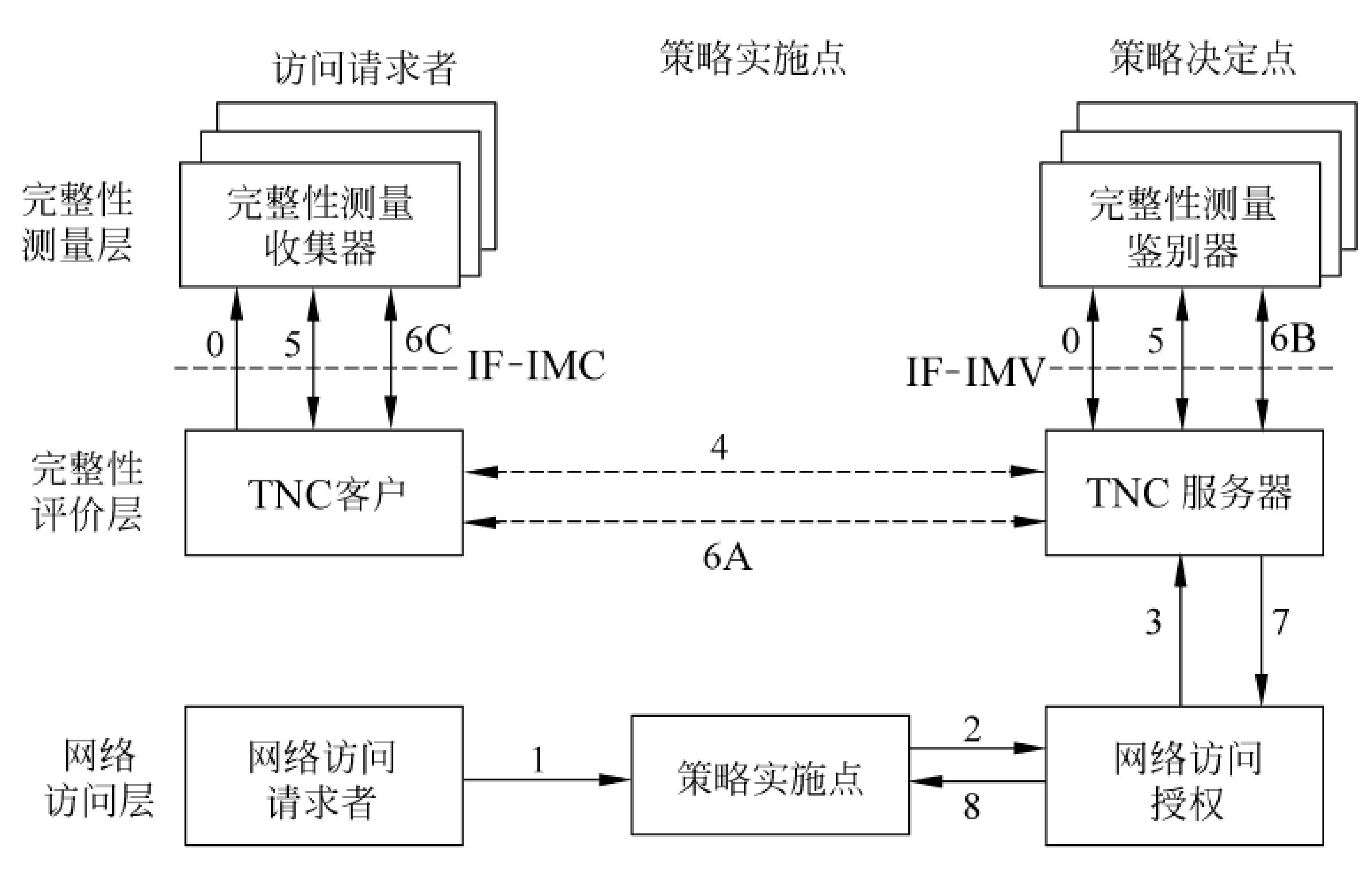


图 1.4.8 TNC 信息流动

(1) 信息 0：在开始网络连接和完整性检查握手协议之前，TNCC 必须装载每一个相关的 IMC，然后启动这些 IMC。还要保证与这些 IMC 连接的状态是不可破坏的。类似地，TNCS 也要装载并启动 IMV。

(2) 信息 1：当有网络连接请求发生时，NAR 在网络层启动一个连接请求。

(3) 信息 2：收到网络连接请求之后，PEP 发送一个网络访问决定请求给 NAA。这里假定 NAA 已经设置成按照用户认证、平台认证和完整性检查的顺序进行操作。如果有一个认证失败，则其后的认证将不会发生。用户认证可以发生在 NAA 和 AR 之间。平台认证和完整性检查发生在 AR 和 TNCS 之间。

(4) 信息 3：假定 AR 和 NAA 之间的用户认证成功完成，则 NAA 通知 TNCS 有一个网络连接请求到来。

(5) 信息 4：TNCS 和 TNCC 进行双向的平台认证。

(6) 信息 5：TNCS 和 TNCC 之间的双向平台认证完成之后，TNCS 通知 IMV 有一个

网络连接请求到来,需要执行一个完整性检查握手。相似地,TNCC 也通知 IMC 有一个网络连接请求到来,需要执行一个完整性检查握手。

(7) 信息 6A:为了执行一个完整性检查握手,TNCS 和 TNCC 开始交换与完整性检查相关的各种信息。这些信息将会被 NAR,PEP 和 NAA 转发,直到 AR 的完整性状态满足 TNCS 的要求为止。

(8) 信息 6B:TNCS 将每个 IMC 信息发送给相应的 IMV,IMV 对 IMC 信息进行分析。如果 IMV 需要更多的完整性信息,它将通过 IF-IMV 接口向 TNCS 发送信息。如果 IMV 已经对 IMC 的完整性信息做出判断,它将结果通过 IF-IMV 接口发送给 TNCS。

(9) 信息 6C:相似地,TNCC 也要转发来自 TNCS 的信息给相应的 IMC,并将来自 IMC 的信息发送给 TNCS。

(10) 信息 7:当 TNCS 完成与 TNCC 的完整性检查握手之后,它发送 TNCS 动作建议给 NAA。这里需要注意的是,即使 AR 通过了 TNCS 的完整性检查,如果它的某些安全属性不满足 NAA 的要求,NAA 仍然可以拒绝 AR 的网络访问请求。

(11) 信息 8:NAA 发送网络访问决定给 PEP 来具体实施。NAA 也必须向 TNCS 说明它最后的网络访问决定,这个决定也将会发送给 TNCC。通常,PEP 会向 NAR 指示它对网络访问决定的执行情况。

可信网络连接技术规范的实现不需要 TPM 模块。尽管采用基于可信网络连接技术规范的解决方案能够保护网络的安全,使用 TPM 的网络系统可能由于具有更高的安全和信任级别而受益。在构建安全的网络环境的过程中,安全产品作为第一道安全防线,正受到越来越多用户的关注。TNC 架构是一个可信网络安全技术体系,试图通过现有网络安全产品和网络安全子系统的有效管理和整合,并结合可信网络的接入控制机制、网络内部信息的保护和信息加密传输机制,全面提高网络整体安全防护能力。

参考文献

1 Lampson B W. Protection. In: Proceedings of the 5th Princeton Symposium. Information Sciences and Systems, Princeton Univ., Princeton, N. J., 1971-03: 437～443, Reprinted in Operating System Rev., 1974, 8(1): 18～24

2 Bell D E, Lapadula L J. Secure Computer Systems: Mathematical Foundations and Model. Bedford, MA: The Mitre Corporation, 1973

3 Harrison M, Ruzzo W, Ullman. Protection in operating systems. CACM, 1976, 8(19): 461～471

4 Biba K J. Integrity considerations for secure computer system. The MITRE Corporation, Technical Report, N MTR-3153, 1997

5 DoD. Trusted Computer System Evaluation Criteria (TCSEC). DoD 5200.28-STD,1985

6 Ferraiolo D, Kuhn D R. Role-based access control. In: Proceedings of the NIST-NSA National (USA) Computer Security Conference, 1992, 554～563

7 Sandhu R, et al. Role-based access control models. IEEE Computer, 1996, 29(2): 38～47

8 Sandhu R, Bhamidipati V, Munawer Q. The ARBAC97 Model for role-based administration of roles. ACM Trans on Information and System Security (TISSEC), 1999, 2(1): 105～135

9 Sandhu R, Munawer Q. The ARBAC99 Model for administration of roles. ACSAC 1999, 1～10

10 Ferraiolo D F，Sandhu R，Gavrila S. Proposed NIST standard for role-based access control. ACM Trans on Information and Systems Security (TISSEC)，2001，4(3)：224～274
11 Clark D D，Wilson D R. A comparison of commercial and military computer security polices. IEEE Symposium of Security and Privacy，1987. 184～194
12 Thomas R K，Sandhu R S. Task-based authentication control (TBAC)：A family of models for active and enterprise-oriented authentication management. In：Proceedings of the 11th IFIP WG 11. 3 Conference on Database Security，1997. 11～13
13 Sejong O，Seog P. Task-role-based access control model. Information System，2003，28(6)：533～562
14 Park J，Sandhu R. Towards usage control models：Beyond traditional access control. In：Proceedings of the 7th ACM Symposium on Access Control Models and Technologies. 2002. 57～64
15 Park J，Sandhu R. The UCON usage control model. ACM Trans on Information and System Security，2004，7(1)：128～174
16 Sandhu R，Park J. Usage control：A vision for next generation access control. MMM-ACNS 2003，17～31
17 Brewer D F C，Nash M J. The Chinese Wall security policy. In：Proceeding IEEE Computer Society Symposium on Research in Security and Privacy，April 1989. 215～228
18 Moyer M J，Ahamad M. Generalized role-based access control. In：Proceedings of IEEE 21th International Conference on Distributed Systems，2001. 391～398
19 Thomas R K. Team-based access control (TMAC)：A primitive for applying role-based access controls in collaborative environments. In：Proceedings of the 2nd ACM Workshop on Role-Based Access Control，Fairfax，Virginia，1997. 13～19
20 Georgiadis C K，Mavridis L，Pangalos G，Thomas R K. Flexible team-based access control using contexts. In：Proceedings of the 6th ACM Symposium on Access Control Models and Technologies，2001. 21～27
21 Alotaiby F T，Chen J X. A model for team-based access control (TMAC 2004). In：Proceedings of the International Conference on Information Technology：Coding and Computing，2004. 450～454
22 Giuri L，Iglio P. Role templates for content-based access control. In：Proceedings of the 2nd ACM Workshop on Role-Based Access Control，Fairfax，Virginia，1997. 153～159
23 Yamazaki W，Hiraishi H，Mizoguchi F. Designing an agent-based RBAC system for dynamic security policy. In：Proceedings of the 13th IEEE International Workshops on Enabling Technologies：Infrastructure for Collaborative Enterprises，2004. 199～204
24 Bertino E，Ferrari E，Atluri V. The specification and enforcement of authorization constraints in workflow management system. ACM Trans on Information and System Security，1999，2(1)：65～104
25 Ahn G，Sandhu R. Role-based authorization constraints specification. ACM Trans on Information and System Security，2000，3(4)：207～226
26 Ahn G，Sandhu R. The RSL99 language for role-based separation of duty constraints. In：Proceedings of the 4th ACM Workshop on Role-based Access Control，1999，New York，43～54
27 Bertino E，Bettini C，Ferrari E，Samarati P. An access control model supporting periodicity constraints and temporal reasoning. ACM Trans on Database System，1998，23(3)：231～285
28 Bertino E，Bonatti P A，Ferrari E. TRBAC—A temporal role-based access control. ACM Trans on Information and System Security，2001，4(3)：191～223
29 Atluri V，Gal A. An authorization model for temporal and derived data：Securing information Portals. ACM Trans on Information and System Security，2002，5(1)：62～94

30 Joshi J B D, Bertino E, Latif U, Ghafoor A. Generalized temporal role-based access control model (GTRBAC) (Part Ⅰ)—Specification and modeling. Technical Report CERIAS TR 2001-47, Purdue University, 2001

31 Joshi J B D, Bertino E, Latif U, Ghafoor A. A generalized temporal role-based access control model. IEEE Trans on Knowledge and Data Engineering, 2005, 17(1): 4～23

32 Bishop M. Vulnerabilities analysis. In: Proceedings of the Second International Symposium on Recent Advances in Intrusion Detection, Sept. 1999. 125～136

33 Varadharajan V. Petri net based modeling of information flow security requirements. In: Proceedings of Computer Security Foundations Workshop Ⅲ, 1990. 51～61

34 Marc D. A Petri net representation of the Take-Grant model. In: Proceedings of Computer Security Foundations Workshop Ⅵ, 1993. 99～108

35 Dong X, Chen G. Petri-net-based context-related access control in workflow environment. In: Proceedings of the 7th International Conference on Computer Supported Cooperative Work in Design, 2002. 381～384

36 Atluri V, Huang W K. A Petri net based safety analysis of workflow authorization models. Journal of Computer Security, 2000, 8(2/3)

37 Knorr K. Dynamic access control through Petri net workflows. Computer Security Applications, ACSAC '00. 16th Annual Conference, Dec. 2000. 159～167

38 Knorr K. Multilevel security and information flow in Petri net workflows. In: Proceedings of the 9th International Conference on Telecommunication Systems-Modeling and Analysis, Special Session on Security Aspects of Telecommunication Systems, Dallas, TX, March. 2001. 9～20

39 Knorr K, Röhrig S. Security requirements of electronic business processes. In: Proc of the First IFIP Conference on E-Commerce, E-Business, and E-Government (I3E), Zurich, Oct. 2001. 73～86

40 Knorr K. Security in Petri Net Workflows. Ph. D thesis, University of Zurich, Switzerland, 2001

41 Bell D, LaPadula L. The Bell-LaPadula model. Journal of Computer Security, 1996, 4 (2, 3): 239～263

42 Jensen K. Colored Petri Nets: Basic Concepts, Analysis Methods and Practical Use, Vol. 1. 2nd edn. Berlin, Germany: Springer-Verlag, 1997

43 Myers C, Clack C, Poon E. Programming with Standard ML. New York: Prentice Hall, 1993

44 Girault C, Valk R. Petri Nets for System Engineering: A Guide to Modeling, Verification and Application. Springer-Verlag, 2003

45 Clarke E M, Grumberg O, Peled D. Model Checking. Cambridge, MA: MIT Press, 2001, 35～49

46 Johnson D, Perkins C, Arkko J. Mobility support in IPv6. RFC 3775, June 2004

47 Perkins C. IP mobility support. RFC 2002, October 1996

48 Soliman H, Catelluccia C, El Malki K, Bellier L. Hierarchical mobile IPv6 mobility management (HMIPv6). RFC 4140, August, 2005

49 Basile C, Powell D. A survey of dependability issues in mobile wireless networks. Technical Report, LAAS CNRS Toulouse, France, 2003

50 Thomson S, Narten T. IPv6 stateless address auto configuration. RFC 2462, December 1998

51 Droms R, Bound J, Bolz B, Lemon T, Perkins C, Carney M. Dynamic host configuration protocol for IPv6 (DHCPv6). RFC 3315, July 2003

52 R. Koodli. Fast handovers for mobile IPv6. RFC 4068, July 2005

53 Kent S. Security architecture for the Internet protocol. RFC 2401, November 1998

54 林闯，彭学海.可信网络研究.计算机学报，2005，28(5)：751～758
55 Ranganathan K. Trustworthy pervasive computing: The hard security problems. In: Proceedings of the 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004, Orlando, FL, 117～121
56 Recovery Oriented Computing. http://www.stanford.edu, or http://roc.cs.berkeley.edu
57 Gates B. Trustworthy computing. http://www.wired.com/news/business/0,1367,49826,00.html
58 Algridas A, Laprie J C, Brian R, Carl L. Basic concepts and taxonomy of dependable and secure computing. IEEE Trans on Dependable and Secure Computing, 2004, 1(1): 11～33
59 邢栩嘉，林闯，蒋屹新. 基于网络的计算机脆弱性评估，计算机学报，2004，27(1)：1～11
60 Nicol D M, Sanders W H, Trivedi K S. Model-based evaluation: From dependability to security. IEEE Trans on Dependable and Secure Computing, 2004, 1(1): 48～65
61 Clark D. A new vision for network architecture. http://www.isi.edu/know-plane/DO-CS/DDC_knowledgePlane_3.pdf
62 Paulson L D. Stopping intruders outside the gates. IEEE Computer, 2002, 35(11): 20～22
63 International Standards Organization. Information processing systems—OSIRM. Part 2: Security architecture, ISO/TC 97 7498-2, 1998
64 Ellison R J. Fisher D A. Survivability—A new technical and business perspective on security. http://www.cert.org/archive/pdf/busperspec.pdf
65 Ellison R J, Fisher D A, Linger R C, et al. Survivable network systems: An emerging discipline. http://www.cert.org/research/97tr013.pdf
66 Vaidya N H. A case for two-level recovery schemes. IEEE Trans on Computers, 1998, 47(6): 656
67 Avizienis A. Design of fault-tolerant computers. In: Proc of AFIPS Conference, 1967, 31, 733～743
68 Michael A, Partha P, et al. Adaptive cyberdefense for survival and intrusion tolerance. IEEE Internet Computing, 2004, 8(6): 25～33
69 Valdes A, Almgren M, et al. An architecture for an adaptive intrusion-tolerant server. http://www.csl.sri.com/papers/p/r/protocols_SRI_02/protocols_SRI_02.pdf
70 林闯，任丰源. 可控可信可扩展的新一代互联网，软件学报，2004，15(6)：960～967
71 Anderson R J. Cryptography and competition policy—Issues with trusted computing. In: Proc of PODC'03, Boston, Massachusetts, USA, July 13—16, 2003
72 Vaughan-Nichols S J. How trustworthy is trusted computing. IEEE Computer, 2003, 36(3): 18～20
73 Felten E W. Understanding trusted computing: Will its benefits outweigh its drawbacks? IEEE Security & Privacy Magazine, 2003, 1(3): 60～62
74 Neumann P G. Principled assuredly trustworthy composable architectures. http://www.csl.sri.com/neumann/chats4.html
75 Ellison R J, Moore A P. Trustworthy refinement through intrusion-aware design: An overview. http://www.cert.org/archive/pdf/triad.pdf
76 Yumerefendi A R, Chase J S. Trust but verify: Accountability for network services. http://issg.cs.duke.edu/publications/trust-ew04.pdf
77 Camp L J. Designing for trust. Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), 2003. 15～29
78 Basile C, Killijian M-O, Powell D. A survey of dependability issues in mobile wireless networks. Technical Report, LAAS CNRS Toulouse, France, 2003. http://www.cr-hc.uiuc.edu/～basilecl/papers/mobile.ps

79 Ganeriwal S, Srivastava M B. Trustworthy sensor networks: Issues & challenges. NESL Technical Report, August 2004. http: //www.ee.ucla.edu/～saurabh/publications/tech- report-integrity.pdf
80 Theodorakopoulos G, Baras J S. Trust evaluation in ad-hoc networks. In: Proceedings of the 2004 ACM Workshop on Wireless Security, WiSe, 2004. 1～10
81 Holly X. Trust management for mobile IPv6 binding update. In: Proc of the International Conference on Security and Management, Las Vegas, NV, USA, 2003. 469～474
82 Mcknight D, Chervany N. The meanings of trust. Carlson School of Management, University of Minnesota, Technical Report TR94-04, 1996
83 Liu J, Issarny V. Enhanced reputation mechanism for mobile ad hoc networks. In: Proceedings of the 2nd International Conference of Trust Management, Oxford, UK, 2004
84 Michiardi P, Molva R. CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad-hoc networks. In: Proceedings of the 6th Joint Working Conference on Communications and Multimedia Security, 2002
85 Marti S, Giuli T, Lai K, Baker M. Mitigating routing misbehavior in mobile ad-hoc networks. In: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Boston, MA, 2000. 255～265
86 Buchegger S, Le Boudec J-Y. Performance analysis of the CONFIDANT protocol. In: Proceedings of the 3rd ACM International Symposium on Mobile Ad-Hoc Networking and Computing, Lausanne, CH, 2002
87 Wang Y, Vassileva J. Trust and reputation model in peer-to-peer networks. In: Proceeding of the 3rd International Conference on Peeer-to-Peer Computing, September, 2003. 150～157
88 Wang Y, Vassileva J. Bayesian network trust model in peer-to-peer networks. In: Proceedings of the 2nd International Workshop on Agents and Peer-to-Peer Computing. Berlin: Springer-Verlag, 2004. 23～34
89 Adams W J, Davis N J. Toward a decentralized trust-based access control system for dynamic collaboration. In: Proceedings of the IEEE Workshop on Information Assurance and Security, United States Military Academy, West Point, NY, 2005. 317～324
90 Griffiths N. Task delegation using experience-based multi-dimensional trust. In: Proceedings of the 4th International Joint Conference on Autonomous Agents and Multi-Agent Systems, 2005. 489～496
91 Srivatsa M, Xiong L, Liu L. TrustGuard: Countering vulnerabilities in reputation management for decentralized overlay networks. In: Proceedings of the 14th World Wide Web Conference, Chiba, Japan, 2005. 422～431
92 Aberer K, Despotovic Z. Managing trust in a peer-to-peer information system. In: Proceedings of the 10th International Conference on Information and Knowledge Management, Atlanta, GA, 2001
93 Cornelli F, Damiani E, De Capitari di Vimercati S, Paraboschi S, Samarati P. Choosing reputable servents in a P2P network. In: Proceedings of the 11th World Wide Web Conference, Hawaii, USA, 2002
94 Kamvar S D, Schlosser M T, Molina H G. The EigenTrust algorithm for reputation management P2P networks. WWW2003, Budapest, Hungary, 2003
95 林闯，封富君，李俊山. 新型网络环境下的访问控制技术. 软件学报,2007,18(4): 955～966
96 Trusted Computing Group, Trusted Computing Platform Alliance (TCPA) Main Specification Version 1.1b, February 2002
97 Trusted Computing Group. TCG Trusted Network Connect TNC Architecture for Interoperability Specification Version 1.0. 2005

Chapter

第 2 章

认证

认证(authentication)是指用户在使用网络系统中的资源时对用户身份的确认。这一过程通过与用户的交互获得身份信息(如用户名/口令组合、生物特征等),然后提交给认证服务器,后者对身份信息与存储在数据库里的用户信息进行核对处理,根据处理结果确认用户身份是否正确。

本章的 AAA 服务器是为流媒体系统设计的,完成接入认证、授权以及计费功能。目前,由于 RADIUS 协议仍然是唯一的 AAA 协议标准,因此本章中设计的 AAA 服务器仍采用 RADIUS 协议,实现 RADIUS 协议中提供的 AAA 服务功能,同时提供用户和计费信息的存储与管理等功能。此外,本章还讨论了多级安全域的认证模型和 DoS 攻击容忍的认证模型,并给出了模型的安全分析。

2.1 RADIUS 协议

RADIUS(remote authentication dial-in user service,远程认证拨号用户服务)的最初设计是为了管理通过串口和调制解调器上网的大量分散用户,后来人们对它进行扩充和完善,使得该协议广泛应用于用户的接入管理,成为当今最流行的用户接入管理协议,为网络提供目前最成熟的用户身份认证(authentication)、授权(authorization)和计费(accounting)功能,即 AAA 管理。

1997 年 1 月,RADIUS 协议问世,因其结构良好、实现简单、扩展灵活等特点引起人们的浓厚兴趣与关注。三个月后,RFC 2138[1] 和 RFC 2139[2] 草案产生。1998 年 12 月,IETF 在第 43 次会议上成立了 AAA 工作组,着手 AAA 相关标准的研究,讨论关于认证、授权和计费的问题。2000 年 6 月,RFC 2865[3] 和 RFC 2866[4] 对 RADIUS 协议进行了进一步的改进和完善,使 RADIUS 协议成为一项通用的 AAA 协议,在 ADSL 接入、以太网接入、无线网络接入等领域中得到广泛应用,成为目前最常用的 AAA 协议之一。但是 RADIUS 协议仍然有不少可以改进之处,比如简单的丢包机制、没有关于重传的规定和集中式计费服务。这些问题使得它不太适应当前网络的发展,需要进一步改进。2000 年开始对 RADIUS 进行深入讨论,提出 RFC 2867[5] 和 RFC 2868[6]。2003 年,IETF 的 AAA 工作组再次从根本上对 AAA 体系结构进行了讨论,提出 RFC 3575[7]。

2.1.1　RADIUS 协议简介

2.1.1.1　RADIUS 协议的主要特点

RADIUS 是应用层协议，基于 UDP 协议。RADIUS 认证使用 1812 端口[3]，计费使用 1813 端口[4]。

概括来说，RADIUS 的主要特点如下：

（1）客户端/服务端模式（client/server）

RADIUS 是一种 C/S 结构的协议，它的客户端最初就是网络接入服务器（network access server，NAS），现在运行在任何硬件上的 RADIUS 客户端软件都可以成为 RADIUS 的客户端。客户端的任务是把用户信息（用户名/密码）传递给指定的 RADIUS 服务器，并负责处理返回的响应。

RADIUS 服务器负责接收用户的连接请求，对用户身份进行认证，并为客户端返回所有为用户提供服务所必需的配置信息。一个 RADIUS 服务器可以为其他 RADIUS 服务器或其他认证服务器担当代理。

（2）网络安全

客户端和 RADIUS 服务器之间的交互经过了共享保密字的认证。另外，为了避免某些人在不安全的网络上监听获取用户密码的可能性，在客户端和 RADIUS 服务器之间的任何用户密码都是加密后传输的。

（3）灵活的认证机制

RADIUS 服务器可以采用多种方式来认证用户的合法性。当用户提供了用户名和密码后，RADIUS 服务器可以支持点对点的 PAP 认证（PPP PAP）、点对点的 CHAP 认证（PPP CHAP）、UNIX 的登录操作（UNIX Login）或其他认证机制。

（4）扩展协议

所有的交互都包括可变长度的属性字段。为满足实际需要，用户可以加入新的属性值。新的属性值可以在不中断已存在协议执行的前提下自行定义新的属性。

2.1.1.2　RADIUS 协议分组格式

RADIUS 数据分组必须遵循如图 2.1.1 所示的格式。在 RADIUS 数据分组中，有 Code（代码），Identifier（标识符），Length（长度），Authenticator（认证码），Attribute（属性）5 个字段域，每个域都按照从左到右的顺序在网络中传送。

1. Code 字段

Code 字段占一个字节长度，标识 RADIUS 消息分组类型。如果收到的分组中代码字段无效，则简单地丢弃该消息。RADIUS 代码值（十进制）具体分配如下：

1　接入请求（access-request）

2　接入允许（access-accept）

3　接入拒绝（access-reject）

4　计费请求（accounting-request）

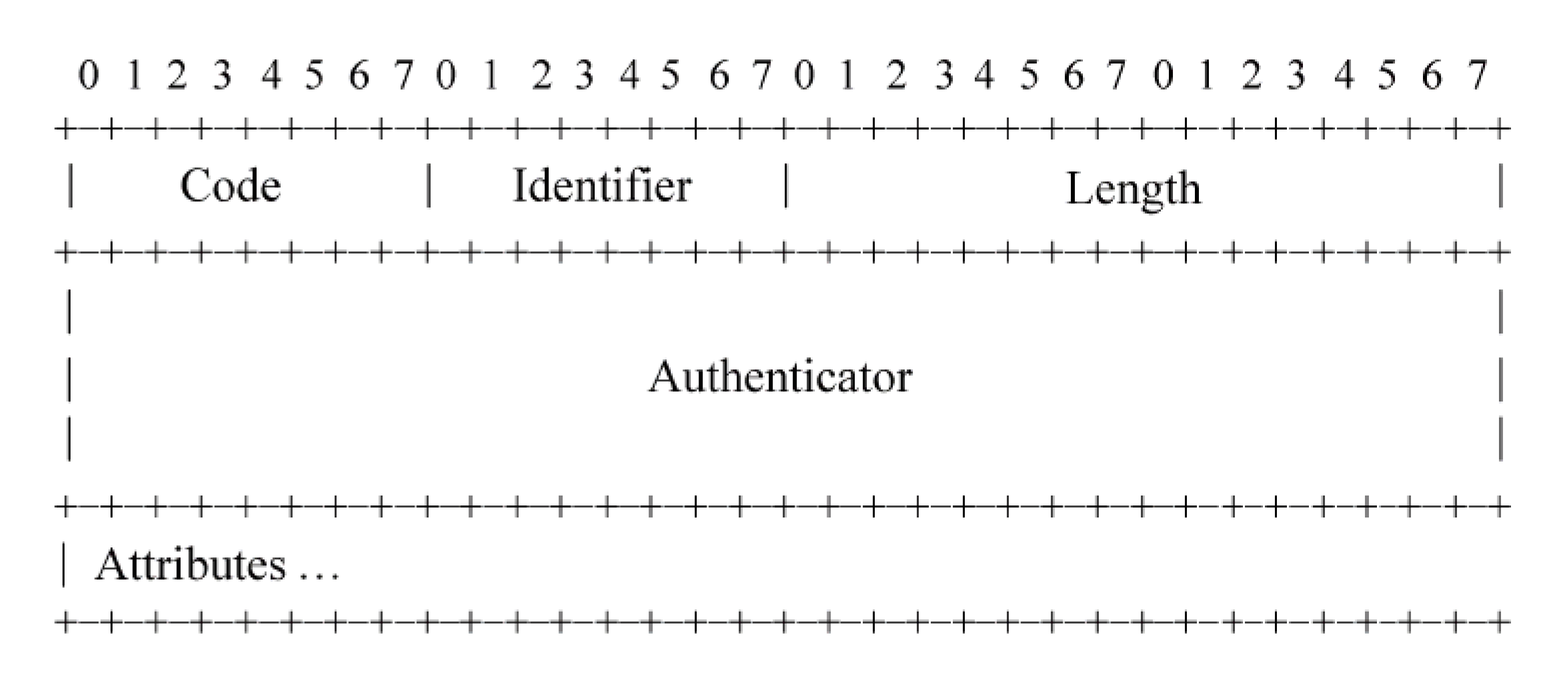


图 2.1.1 RADIUS 数据包格式

5 计费响应(accounting-response)

11 接入询问(access-challenge)

12 服务器状态(status-server (experimental))

13 客户机状态(status-client (experimental))

255 预留(reserved)

2. Identifier 字段

Identifier 字段占一个字节长度,一般来说是一个短期内无法重复的数字,用于匹配请求与应答。RADIUS 服务器能检测出具有相同的客户源 IP 地址、源 UDP 端口及标识符的重复请求。

3. Length 字段

Length 字段占两个字节长度,它指的是包含代码、标识符、长度、认证者和属性域的分组总长度。超出长度域所指示的部分将被看作是填充字节而被忽略接收。如果分组长度比长度域所指示的短,则必须丢弃该分组。长度值最小为 20 字节,最大为 4096 字节。

4. Authenticator 字段

Authenticator 字段占 16 个字节,用于口令隐藏算法,同时能够认证 RADIUS 服务器的应答。认证码有请求认证码和响应认证码两种。

(1) 请求认证码(request authenticator):在接入请求(access-request)数据包中,认证码值是一个 16 个字节的随机二进制数,称为请求认证码。值得注意的是,在密钥的整个生存周期中,这个值应该是不可预测的,并且是唯一的,因为具有相同密钥的重复请求值,使黑客有机会用已截取的响应回复用户。因为同一密钥可以被用在不同地理区域中的服务器的验证中,所以请求认证域应该具有全球和临时唯一性。另外,在请求接入和请求计费协议包中的请求认证码的生成方式是有区别的。对于请求接入包,请求认证码是 16 个 8 位字节的随机数。对于计费请求包,认证码是一串由(Code+Identifier+Length+16 个为 0 的 8 位字节+请求属性+共享密钥)所构成的字节流经过 MD5 加密算法计算出的散列值。

(2) 响应认证码(response authenticator):响应认证码是接入允许、接入拒绝、接入询问和计费响应数据包中的认证码值,它包含了在一串字节流上计算出的单向 MD5 散列,这些二进制数由 RADIUS 数据包组成,包括编码域、标识符、长度以及来自接入请求数据包的

请求认证码和执行共享机密的响应属性。即：

ResponseAuth＝MD5(Code＋ID＋Length＋RequestAuth＋Attributes＋Secret)

5. Attribute 字段

Attribute 字段为可变长度，不同类型的分组其属性字段的内容和取值不同。RADIUS 消息的长度字段值指明了属性列表的结束。

2.1.1.3 RADIUS 协议中的属性

RADIUS 消息中最重要的就是其属性字段。RADIUS 协议通过不同的属性来实现各种操作的定义，因为不同含义的属性携带不同的信息。认证属性携带认证请求与应答的详细认证、授权信息和配置细节。计费属性携带详细的计费信息。

RADIUS 消息中的各个属性没有先后顺序关系。每个属性有一个代码标识，属性的基本格式如图 2.1.2 所示。

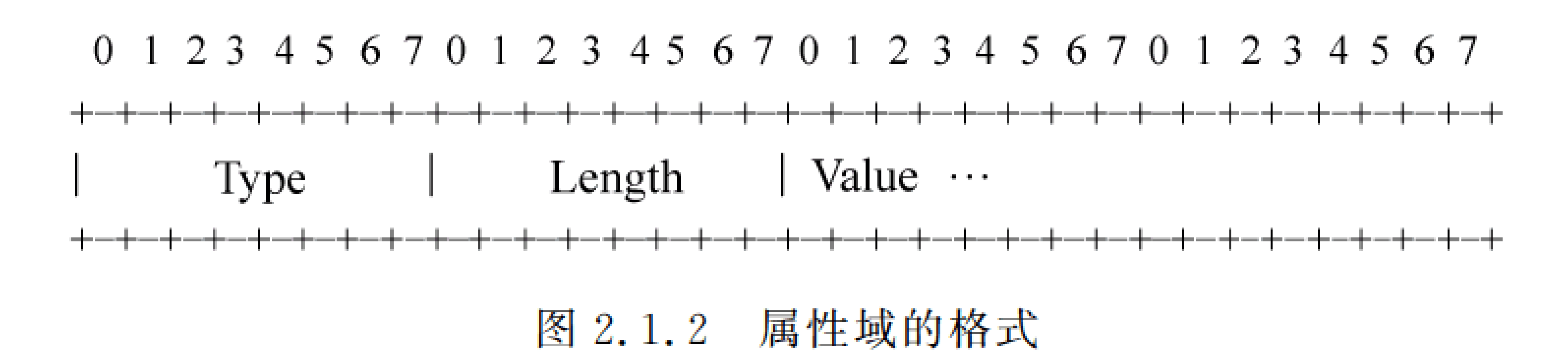


图 2.1.2 属性域的格式

1. 类型(type)

由 1 个字节表示，取值为 1～255。目前分配的范围为 1～63，具体内容在 RFC 2865、RFC 2866 中进行了说明。此外，为了在 RADIUS 协议中封装 EAP(PPP extensible authentication protocol，PPP 的扩展认证协议)包，RFC 2869 定义了两个新的属性：EAP-Message(79)和 Message-Authenticator(80)，其中 EAP-Message 用于封装 EAP 包，而 Message-Authenticator 包含消息摘要以防止 EAP 包被篡改。RADIUS 服务器和客户端都可以忽略不可辨识类型的属性。

2. 长度(length)

由一个字节表示，它指定了包括类型、长度和值域在内的属性长度。如果在接收到的接入请求中属性的长度是无效的，应该发送一个接入拒绝数据包。如果在接收到的接入允许、接入拒绝和接入询问中属性的长度是无效的，该数据包必须处理为接入拒绝，或者直接丢弃。

3. 属性值(value)

可以为 0 或者多个字节，包括属性的详细信息。值域的格式和长度由属性的类型和长度决定。

特别值得一提的是 26 号属性 Vendor-Specific，它用于 NAS 厂商对 RADIUS 进行扩展，以实现标准 RADIUS 协议未定义的功能，如 VPN 等。此属性禁止对 RADIUS 协议中的操作有影响。当服务器不具备解释由客户端发送过来的供应商特性信息的能力时，则服务器必须忽略这些信息。

2.1.2 RADIUS 的安全处理

2.1.2.1 RADIUS 支持的认证操作

标准 RADIUS 协议只规范了 NAS 与 RADIUS 服务器之间交互操作的内容，而对用户主机与 NAS 之间的交互操作未作任何规定和限制，所以，由用户主机与 NAS 协商来决定他们之间使用何种协议。

标准 RADIUS 协议中描述了在用户、NAS、RADIUS 服务器三者之间进行的两种基本认证操作模式：请求/响应模式和质询/应答模式。对应着用户与 NAS 之间使用密码认证协议(password authentication protocol，PAP)和挑战-握手认证协议(challenge-handshake authentication protocol，CHAP)[8]。

对于 PAP 认证，NAS 将用户名和密码作为明文传输给 RADIUS 服务器，RADIUS 根据用户和密码对用户进行认证。如果认证通过，则发送接入允许的包；如果认证未通过，则发送接入拒绝包。

对于 CHAP 认证，NAS 产生一个 16 位的随机码传送给用户，用户端得到这个随机码之后对传过来的数据进行加密，生成一个响应数据包传给 NAS。数据包包含 CHAP ID 和对随机数加密后的数据。NAS 收到这个响应之后，加上原先的 16 位随机码，一起传送给 RADIUS 服务器。服务器收到这个请求包之后，查询数据库找出匹配项与认证服务器相比较，若不满足，则发送接入拒绝包；若满足，则取出随机数和用户共享的加密密码，对随机数采用同样的加密得出一个数据和 NAS 传送过来的数据相比较，若一致，则认证通过，否则拒绝接入。

请求/响应模式操作简单，但因为用户的口令等认证信息要在网络中传输，容易被窃听，安全性较差。而质询/应答模式就不存在这种缺陷，因为用户的口令信息不在网络中传输，而是通过随机产生的质询值使得每次传输的验证信息都以不同的方式来防止信息被窃听，具有较好的安全性。但是这需要服务器端保存明文密码，用来做相同的加密运算才可以比较出结果。

现在，RADIUS 协议已经扩展可以支持用户与 NAS 之间的多种认证方式[9]，如 EAP[10] 等。

2.1.2.2 用户密码的处理

在传输时，密码是被隐藏起来的。首先在密码的末尾用 nulls 代替填补形成多个 16 个字节的二进制数。单向 MD5 散列是通过一串字节流计算出来的，该字节流由共享密钥和跟随其后的请求认证码组成。这个值与密码的第 1 个 16 个字节段相异或，然后将异或结果放在用户密码属性字符串域中的第 1 组 16 个字节中。如果密码长于 16 个字节，则第 2 次单向 MD5 散列对一串字节流进行计算，该字节流由共享机密和跟随其后的第 1 次异或结果组成。这个散列结果与密码的第 2 组 16 个字节段相异或，然后将异或结果放在用户密码属性字符串域中的第 2 组 16 个字节段中。如果需要，上述计算过程可以重复。每一个异或结果被用于和共享机密一道生成下一个散列，再与下一个密码段相异或，但最大不超过 128 个

字节。

其流程如图 2.1.3 所示，描述如下：

（1）调用共享机密 S 和伪随机 128 位请求认证码 RA。

（2）把密码按 16 个字节为一组划分为 P_1，P_2 等，在最后一组的结尾处用 null 填充以形成一个完整的 16 字节组。

（3）调用已加密的数据组 c_i，b_i是将要用到的中间值。

$$b_1 = \mathrm{MD5}(S + \mathrm{RA}), \quad c_1 = P_1 \text{ 异或 } b_1$$
$$b_2 = \mathrm{MD5}(S + c_1), \quad c_2 = P_2 \text{ 异或 } b_2$$
$$\vdots$$
$$b_i = \mathrm{MD5}(S + c_{i-1}), \quad c_i = P_i \text{ 异或 } b_i$$

（4）密码字符串包含 $c_1 + c_2 + \cdots + c_i$，其中“＋”表示串联。

（5）接收时，这个过程被反过来，从而生成原始的密码。

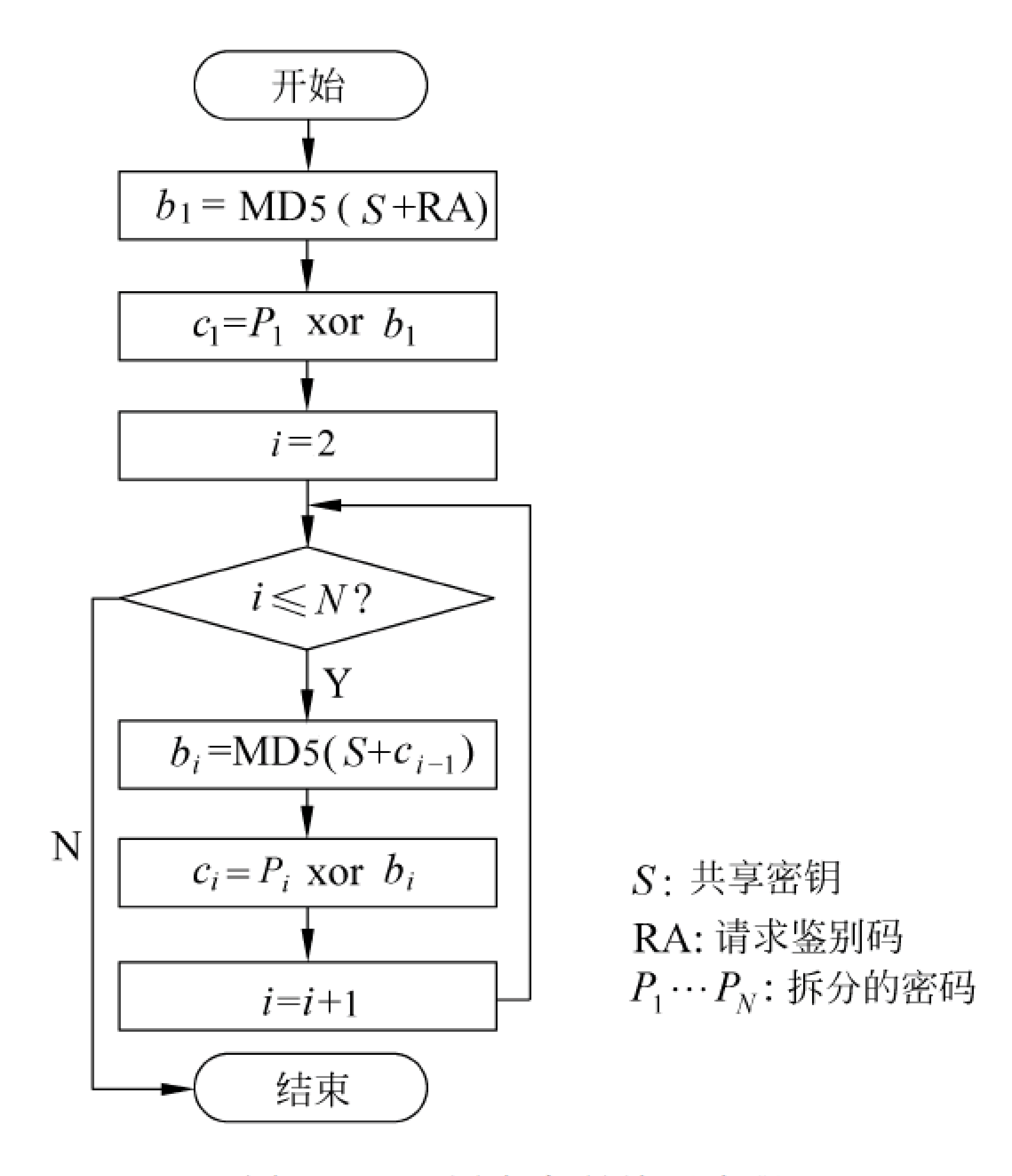


图 2.1.3　用户密码处理流程

2.1.2.3　认证码的处理

RADIUS 数据包中的认证码主要有两个作用：数据包的完整性检查和对客户端的认证。由于相应认证码产生的时候对整个数据包采用共享机密字加密，所以如果这个共享机密字不一致，那么服务器端产生的认证码和 NAS 端传送过来的认证码会不一致。同时，如果数据包在网络中传输的时候有数据丢失，那么两个认证码也会不一致，这样就可以完成数据包的完整性检查。其具体的实现过程如下：

（1）NAS 根据一定的算法，产生 16 个 8 位随机二进制数作为请求认证码，这个值在密码的整个生存周期中是不可预测的，唯一的。

（2）NAS 构造请求接入包，发送给 RADIUS 服务器。

(3) RADIUS 服务器收到 NAS 的接入请求后根据用户名在数据库中查找匹配项，若找到匹配项，采用与客户端一致的方法将用户密码以及与客户端的共享机密字进行加密运算产生认证码。

(4) 用这个运算产生的数据与 NAS 传送过来的认证码加以比较，如果一致，那么认证通过，发送允许接入包，否则发送拒绝接入包。

(5) 服务器构造响应认证码。

(6) 服务器根据上面的响应认证码加上前面的认证结果构造响应包发送给 NAS。

(7) NAS 收到认证应答包后，根据正在等待响应的请求队列中的那个请求，按照刚接收到的应答包的内容和其请求认证码计算一个响应认证码，与 RADIUS 服务器发送过来的这个认证码相比较。若相等，则认证通过，建立连接，否则认证失败。

2.1.2.4 数据包的重传机制

由于 RADIUS 数据包采用 UDP 传输，因此丢失数据包的可能性非常大，协议采用多种措施保证数据传输的可靠性。

(1) 无论是认证请求还是计费请求，在一个指定的时间内没有收到回应，会多次重传，如果超过一定的时间还没有收到响应，那么可以看作主服务器已经关机。这时，NAS 可以选择给一个或者多个备用服务器传送请求。在多次尝试连接主服务器失败后，或在一轮循环方式结束后选择连接后备服务器。

(2) 为保障计费请求的连接，在计费开始的时候要发送一个计费开始请求包，一个呼叫结束之后要发送一个计费结束包。在计费开始请求和计费结束请求中必须包含一个 ACCT_DELAY_TIME 的属性，记录从开始发送请求到计费请求发送出去之间的时间间隔，确保计费信息记录准确。

2.1.3 RADIUS 的工作过程

RADIUS 协议旨在简化认证流程，其典型认证授权工作过程如下：

(1) 用户输入用户名、密码等信息到客户端或连接到 NAS。

(2) 客户端或 NAS 产生一个“接入请求(access-request)”报文到 RADIUS 服务器，其中包括用户名、密码、客户端(或 NAS)ID 和用户访问端口的 ID。口令经过 MD5 算法进行加密。

(3) RADIUS 服务器对用户进行认证。

(4) 若认证成功，RADIUS 服务器向客户端或 NAS 发送允许接入包(access-accept)，否则发送拒绝接入包(access-reject)。

(5) 若客户端或 NAS 接收到允许接入包，则为用户建立连接，对用户进行授权并提供服务，然后转入第(6)步；若接收到拒绝接入包，则拒绝用户的连接请求，结束协商过程。

(6) 客户端或 NAS 发送计费请求包给 RADIUS 服务器。

(7) RADIUS 服务器接收到计费请求包后开始计费，并向客户端或 NAS 回送开始计费响应包。

(8) 用户断开连接，客户端或 NAS 发送停止计费包给 RADIUS 服务器。

(9) RADIUS 服务器接收到停止计费包后停止计费，并向客户端或 NAS 回送停止计费

响应包,完成该用户的一次计费,记录计费信息。

2.2 AAA 服务器设计

2.2.1 AAA 系统概述

AAA 指的是 authentication(认证)、authorization(授权)和 accounting(计费)。自网络诞生以来,认证、授权以及计费体制就成为其运营的基础。网络中各类资源的使用,需要由认证、授权和计费进行管理。而 AAA 的发展与变迁自始至终都吸引着营运商的目光。对于一个商业系统来说,认证是至关重要的,只有确认了用户的身份,才能知道所提供的服务应该向谁收费,同时也能防止非法用户对网络进行破坏。在确认用户身份后,根据用户开户时所申请的服务类别,系统可以授予客户相应的权限。最后,在用户使用系统资源时,需要有相应的设备来统计用户对资源的占用情况,据此向用户收取相应的费用。

其中,认证是指用户在使用网络系统中的资源时对用户身份的确认。这一过程,通过与用户的交互获得身份信息(诸如用户名口令的组合、生物特征等),然后提交给认证服务器;后者对身份信息与存储在数据库里的用户信息进行核对处理,然后根据处理结果确认用户身份是否正确。授权是指网络系统授权用户以特定的方式使用其资源,这一过程指定了被认证的用户在接入网络后能够使用的业务和拥有的权限,如授予的 IP 地址等。计费是指网络系统收集、记录用户对网络资源的使用,以便向用户收取资源使用费用,或者用于审计等目的。

认证、授权和计费一起实现了网络系统对特定用户的网络资源使用情况的准确记录。这样既在一定程度上有效地保障了合法用户的权益,又能有效地保障网络系统安全可靠地运行。

2.2.2 AAA 系统的设计需求

这里的 AAA 服务器是为流媒体系统设计的,完成接入认证、授权以及计费的功能,采用 RADIUS 协议实现 AAA 服务功能,同时系统提供用户和计费信息的存储与管理等功能。该系统需求主要包括以下几个方面:

(1) 用户认证。用户在申请享受服务时,需要得到用户信息的认证。在本系统中,客户端发送 AAA 认证数据包给服务器,数据包包含用户 ID 和 Password,服务器对数据包进行验证给出结果。验证过程中数据包加密传输。

(2) 用户服务授权。不同的用户可以享受不同的服务。AAA 服务器在通过用户的认证请求后,按照该用户的权限来决定用户是否可以享受申请的服务内容。

(3) 服务计费。系统提供基本的计费信息和计费算法,支持一定的计费策略,并保存计费过程产生的中间数据。系统需达到实时计费的要求。计费的最小单位为分,能够保证用户不会透支费用。

(4) 用户信息管理。用户信息管理的主要功能包括用户注册、费用管理查询、权限设置等。用户需要注册才能申请享受服务,用户注册时提供用户名、密码和邮箱等基本资料,且

提供密码遗忘时找回密码的功能。用户可以查询自己费用的详细信息,可以给账户充值。管理员能对注册用户进行管理。

(5) 服务器性能。AAA 系统中需要考虑的服务器性能包括:

① 服务器的可处理容量,包括支持用户数和在某一段时间内支持的并发用户数;

② 可靠性,由于网络原因,数据在传输中常常会丢失,如何减少这种丢失,为认证计费提供尽量可靠的传输是需要考虑的问题;

③ 鲁棒性,即容错性,发生不可避免地丢包时如何保证认证和计费过程的正确性;

④ 请求响应时间,用户在发出请求到收到应答的间隔时间不能太长;

⑤ 最后,对于一个研究中的流媒体平台来说,用户的需求是在不断扩展的。AAA 系统的设计需要充分考虑这一点,意味着系统的可扩展性将非常重要。对于系统的各个模块来说,其部分的改动应该不会影响到其他模块的正常运行,且系统模块的增加应该是容易做到的。

2.2.3 AAA 系统的整体结构

本节介绍的 AAA 系统除包括认证、计费服务器以外,还包括用户和计费信息的存储、用户和计费策略管理等。整体结构如图 2.2.1 所示,系统交互如图 2.2.2 所示。

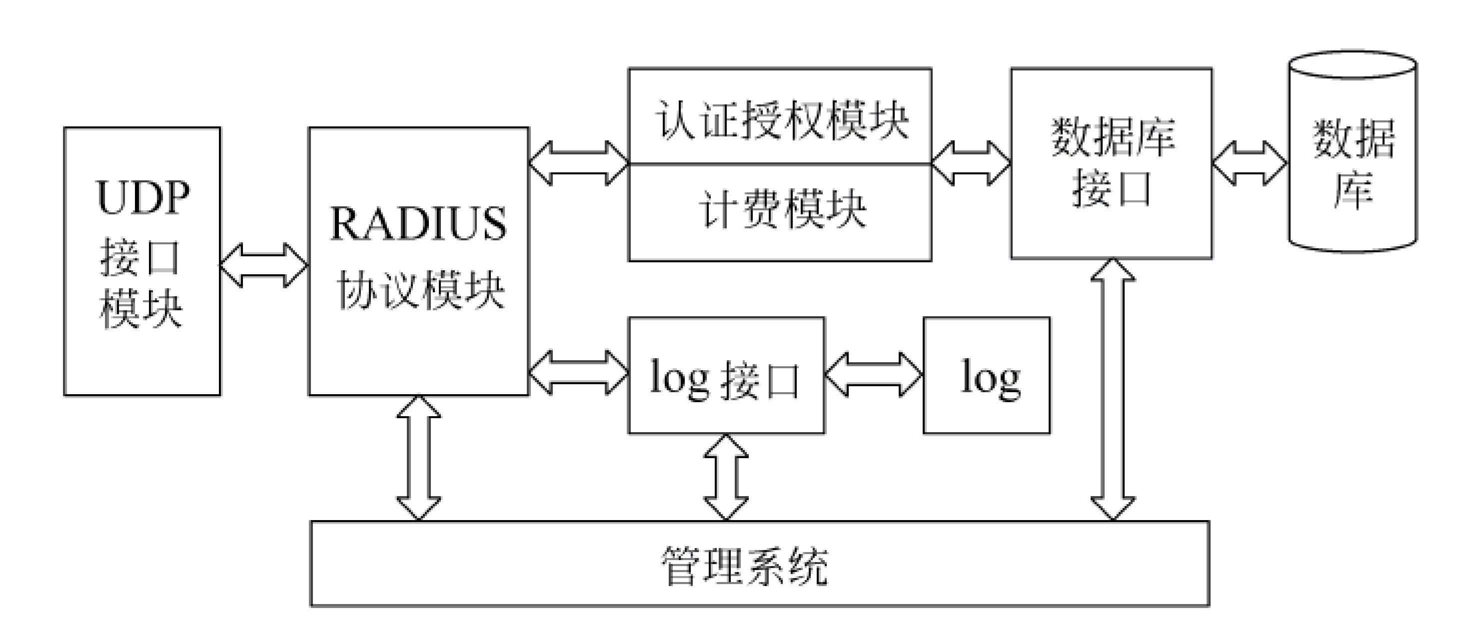


图 2.2.1 AAA 系统功能模块示意图

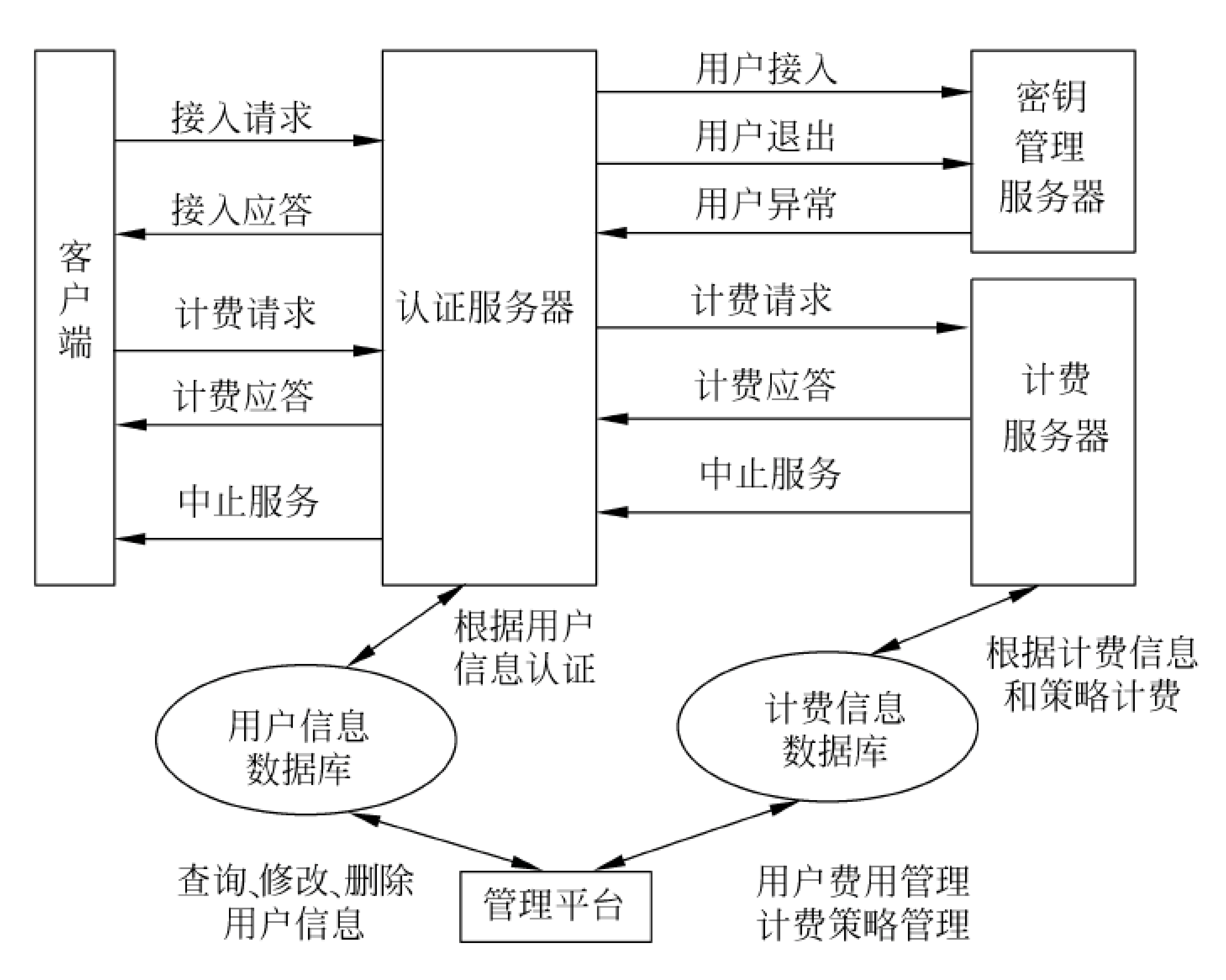


图 2.2.2 AAA 系统交互示意图

考虑到扩展性、计费准确性以及各部分性能要求，我们将 AAA 服务器分为认证/授权和计费服务器两大部分，这种结构为典型的 AAA 系统架构。这种结构可以很容易地扩展为一台认证服务器＋多台计费服务器，或者多台认证服务器＋多台计费服务器的架构，以适合不同规模的流媒体平台应用。

在整个 AAA 系统中，RADIUS 服务器之间以及 RADIUS 认证服务器与 NAS 的通信遵循 RADIUS 协议标准；用户信息和计费信息保存在 MySQL 数据库中，信息管理通过 Web 页面形式进行管理，发布平台采用 PHP＋MySQL 的方式。

2.2.4　AAA 系统的基本设计思想

在流媒体系统中，RADIUS 服务器要处理 5 个方面的内容：用户的认证处理、用户的授权处理、计费开始信号的处理、计费结束信号的处理和中止用户服务信号的处理。服务器大致来说包括 3 个重要的处理模块：收发包处理模块、计费/认证处理模块和代理 Client。其中，收发包处理模块的功能主要是接收 NAS 端发送过来的 RADIUS 数据包，对之作相应的处理，然后把数据包转发给认证/计费处理模块，以及将服务器处理过的数据包按照 RADIUS 协议打包，然后发送到 NAS。

认证/计费处理模块的主要功能是对发送过来的数据包进行认证和计费处理。如果是本地认证，则对数据包直接处理；如果是一个漫游，则向上级服务器转发这个请求。

代理 Client 的主要功能是根据要求，将非本地认证/计费请求按要求转发给相应的上级服务器，同时接受上级服务器处理过的请求，将之转发给收发包处理模块，由收发包处理模块转发到 NAS 终端。

RADIUS 服务器的内部数据处理的流程如图 2.2.3 所示，描述如下：

(1) 收发包处理模块接收到来自 NAS 即 RADIUS Client 的认证/计费请求，将其转交给认证/计费处理模块处理，也就是图 2.2.3 的“请求包 1”的过程；

(2) 如果计费/认证处理模块不能对(1)发送过来的请求包进行处理，则将其作为“请求包 2”转发；

(3) 代理 Client 对“请求包 2”处理，然后作为“请求包 3”向上级转发数据包，请求上级 RADIUS 服务器作响应的计费/认证处理；

(4) “回答包 1”是收发处理模块收到的来自上级的 RADIUS 服务器的应答，转发给代理 Client 处理；

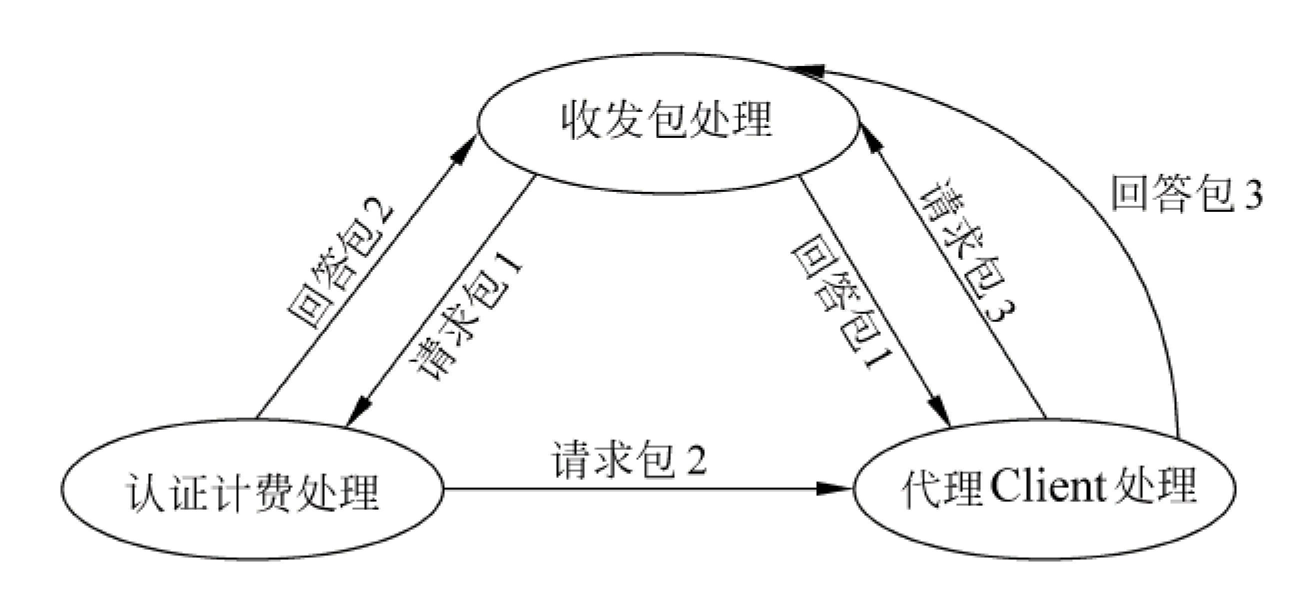


图 2.2.3　服务器内部数据处理流程

(5)“回答包 2”是来自计费/认证处理模块的数据包,是认证/计费处理模块对用户认证/计费的处理结果,发送给收发包处理模块转发给 NAS 的;

(6)“回答包 3”是代理 Client 对上级的 RADIUS 服务器的“应答包 1”处理后交由收发处理模块转发的数据包。

RADIUS 服务器与客户端连接如图 2.2.4 所示。

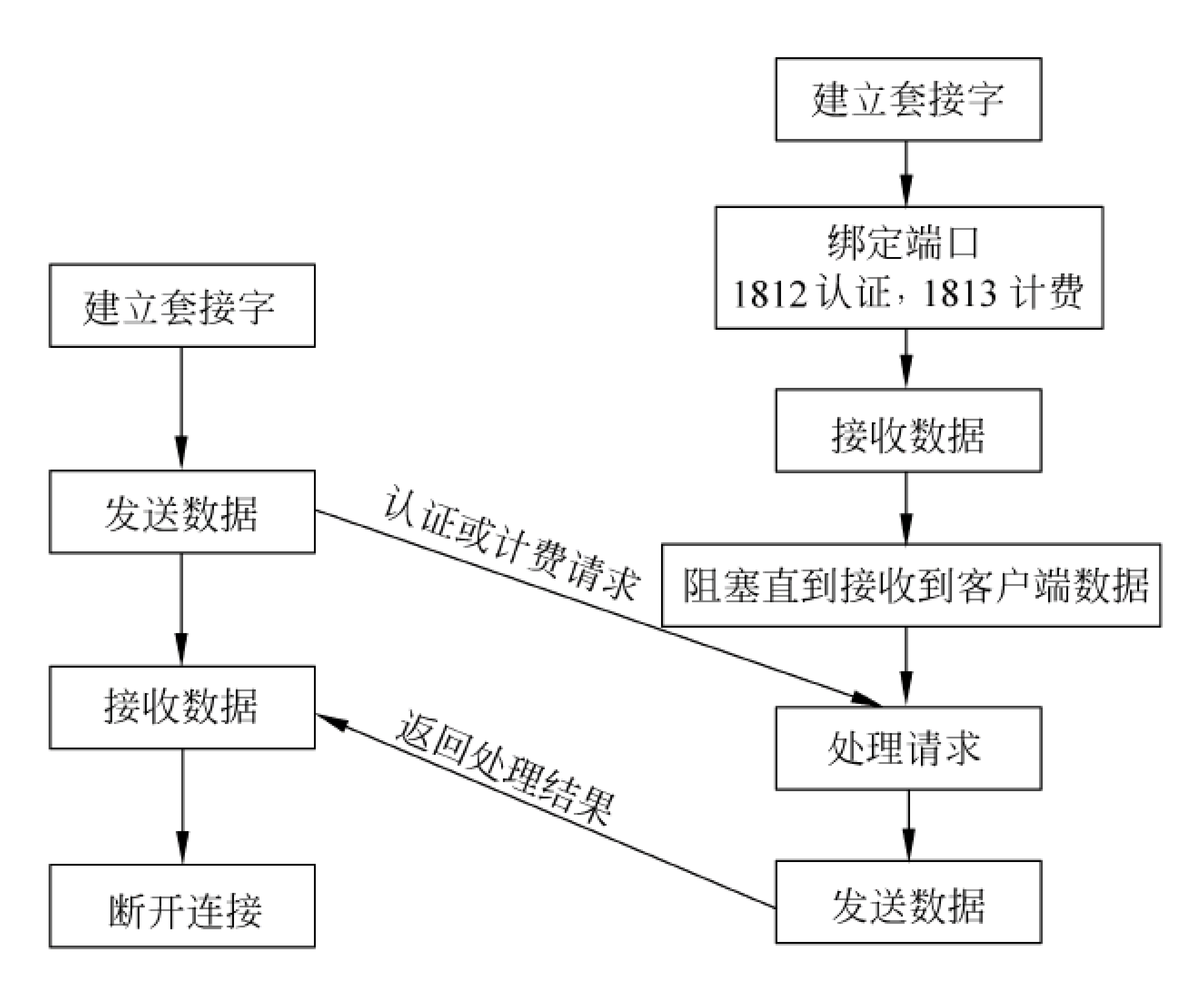


图 2.2.4 RADIUS 服务器与客户端连接示意图

2.2.5 AAA 数据流控制设计

2.2.5.1 服务器软件设计算法

在服务器的控制设计方案中,可以是循环的或并发的,可以使用面向连接的传输或无连接的传输。通常,我们把服务器划分为 4 种类型[11]:无连接循环服务器、面向连接循环服务器、无连接并发服务器以及面向连接并发服务器等。

1. 无连接循环服务器

这是最常见的无连接服务器的形式,特别适用于要求对每个请求进行少量处理的服务。循环服务器往往是无状态的,这使其易于理解而且不易出错。

无连接循环服务器的基本算法如下:

(1)创建套接字并将其绑定到所提供服务的端口上;

(2)重复读取来自客户的请求,构造响应,按照应用协议向客户发回响应。

2. 面向连接循环服务器

这是较常见的服务器类型,它适用于要求对每个请求进行少量处理且要求有可靠的传输。该类型中,建立和终止连接相关的开销可能很高,平均响应时间可能并不短。

面向连接循环服务器的基本算法如下:

(1)创建套接字并将其绑定到所提供服务的端口上;

(2) 将该端口设置为被动模式,使其准备为服务器所用;
(3) 从该套接字上接收下一个连接请求,获得该连接的新的套接字;
(4) 重复地读取来自客户的请求,构造响应,按照应用协议向客户发回响应;
(5) 当与某个特定客户完成交互时,关闭连接,并返回步骤(3)以接收新的连接。

3. 无连接并发服务器

这是一种不常见的服务器类型,服务器要为处理每个请求创建一个新线程或进程。在许多系统中,创建线程或进程所增加的开销决定了由并发性所获得的效率。为证明并发性是可取的,要么创建一个新线程或进程所要求的时间必须明显地小于计算响应所需的时间,要么并发地请求必须能够同时使用多个 I/O 设备。

无连接并发的服务器的最简单的版本遵循下面的算法:

主线程 1. 创建套接字并将其绑定到所提供服务的熟知地址上,让该套接字保持为未连接的;

主线程 2. 反复调用 recvfrom 接收来自客户的下一个请求,创建一个新的从线程(可能在一个新的进程中)来处理响应;

从线程 1. 从来自主进程的特定请求以及到该套接字的访问开始;

从线程 2. 根据应用协议构造应答,并用 sendto 将该应答发回客户;

从线程 3. 退出。

4. 面向连接并发服务器

这是最一般的服务器类型,它提供了可靠的传输以及并发处理多个请求的能力。最常见的实现使用了并发进程或并发线程来处理每个连接,另外可以依赖单线程和异步 I/O 处理多个连接。

无连接循环服务器的基本算法如下:

主线程 1. 创建套接字并将其绑定到所提供服务的熟知地址上,让该套接字保持为非连接的;

主线程 2. 将该端口设置为被动模式,使其准备为服务器所用;

主线程 3. 反复调用 accept 以便接收来自客户的下一个连接请求,并创建新的从线程或进程来处理响应;

从线程 1. 由主线程传递来的连接请求(即针对连接的套接字)开始;

从线程 2. 用该连接与客户进行交互,读取请求并发回响应;

从线程 3. 关闭连接并退出。在处理完来自客户的所有请求之后,从线程就退出。

因为 RADIUS 协议建立在 UDP 之上,故服务器使用无连接循环服务器。其通常做法是采用状态机和流水线来控制数据的流向。

2.2.5.2 RADIUS 状态机

RADIUS 请求包在整个处理过程中分为以下几种状态:空闲、认证/计费、待发送、代理和出错状态。在程序实现时,用线性表来记录包的状态,状态机如图 2.2.5 所示。

(1) 空闲状态。该包中无任何数据。在该状态中,对包不作任何处理。当有"请求包到"事件触发后,状态转为"认证/计费"状态。

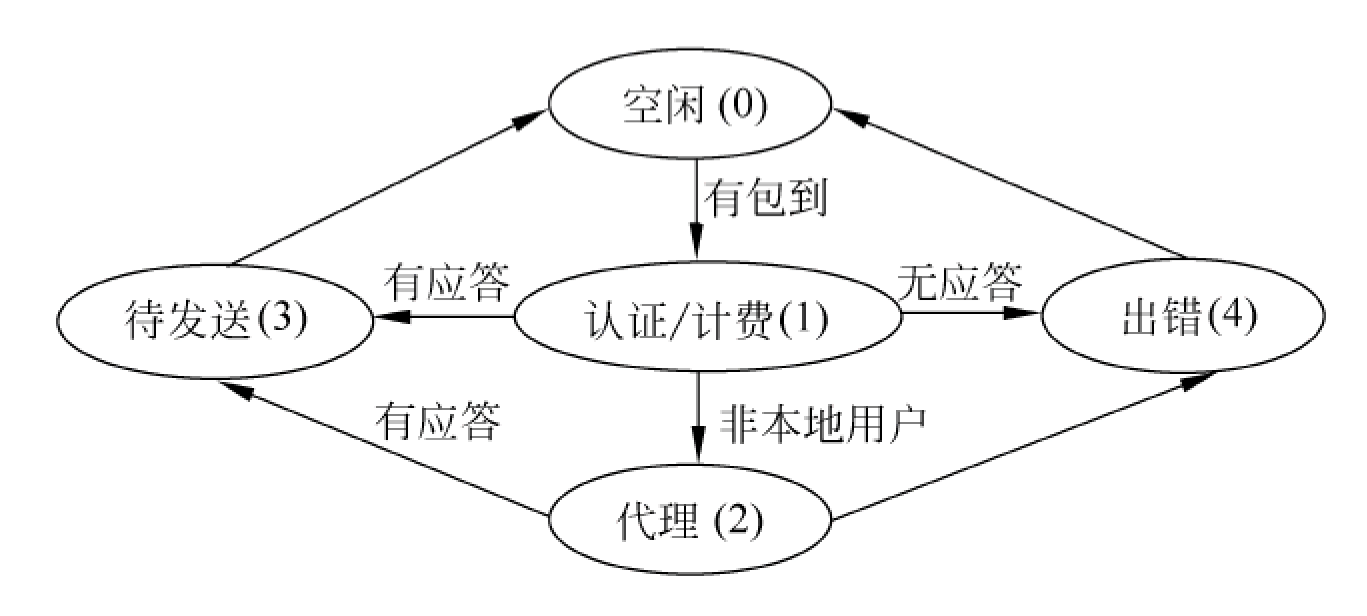


图 2.2.5 RADIUS 服务器处理请求包的状态机

(2) 认证/计费状态。该包是 NAS 或下级 RADIUS 服务器发来的请求包。当处于此状态中的包进行认证/计费处理时,如果数据出错,不需要发应答消息,则变为"出错"状态;如果数据正确,需发回应答消息,组成应答包,则状态变为"待发送"状态。如果认证/计费用户不在本地,即漫游用户,对包需要作代理认证/计费处理,则状态变为"代理 Client"状态。

(3) 代理 Client 状态。该包是经过认证/计费处理过的,认为需要代理认证/计费处理。对该包作代理认证/计费处理,向上级 RADIUS 服务器发送代理请求包,等待上级发回应答,若超时无应答,则状态变为"出错"状态;当有上级发回应答包时,则状态变为"待发送"状态。

(4) 待发送状态。对于该请求包将有应答包发送。向 NAS 或下级 RADIUS 服务器发送应答包,发送完毕,状态变为"空闲"状态。

(5) 出错状态。该包出错,进行出错处理,根据情况给管理员警告消息。处理完转为空闲状态。

2.2.5.3 流水线方案

采用双队列无状态流水线的设计方案,具有以下优点:

(1) 流程清晰,设计简单;

(2) 数据包经流水处理,几乎不存在等待队列的控制权问题,处理速度大为提高;

(3) 对异常情况处理能力有所提高。

双队列无状态流水线的设计方案如图 2.2.6 所示。

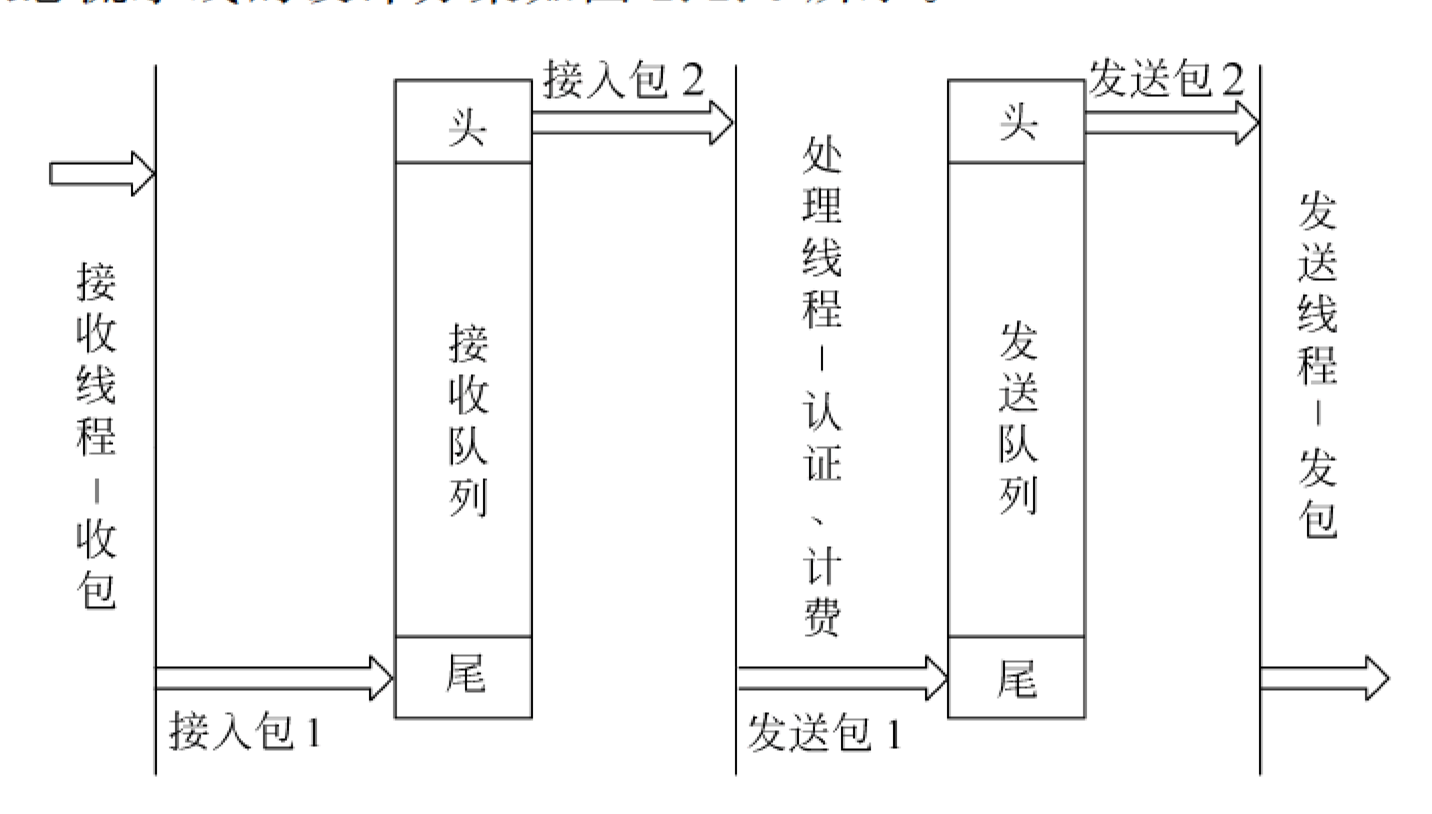


图 2.2.6 服务器流水线设计方案

接入包 1：来自 NAS 中的 RADIUS Client，由“收包处理”加入“接收队列”；

接入包 2：来自“接收队列”，由处理线程取出。如果是请求包，进行认证/计费处理，如果认证/计费服务器不能对请求包完成认证/计费处理，则将其转为发送包 1，向上级 RADIUS Server 发请求，请求上级 RADIUS Server 作认证/计费处理；如果是上级 RADIUS Server 的回答包，则将其转为发送包 1，发回 NAS；

发送包 1：将用户的认证/计费结果和向上级 RADIUS Server 发出的请求包加入“发送队列”，等候发送线程将其发出；

发送包 2：来自“发送队列”，“发送线程”根据目的地址将包发出。

2.2.6 RADIUS 认证服务器

2.2.6.1 总体结构设计

RADIUS 协议本身没有对数据传输作要求，使用 UDP 协议，这使得 RADIUS 协议数据包传输不可靠。数据包的丢包可能发生在网络传输的环节，也可能发生在数据接收端。RADIUS 认证服务器作为 RADIUS 系统对 NAS 的服务前端，必须考虑在 RADIUS 协议包大量并发情况下的性能，包括丢包率、应答延迟时间、待处理数据包排队情况等。为此，RADIUS 服务器需要合理地利用系统资源加以均衡，以避免在服务器大量存在需要处理的数据包的同时 CPU 大量空闲。

认证的流程如图 2.2.7 所示。对于一个认证请求，如果不包括 Proxy 处理，则正常情况

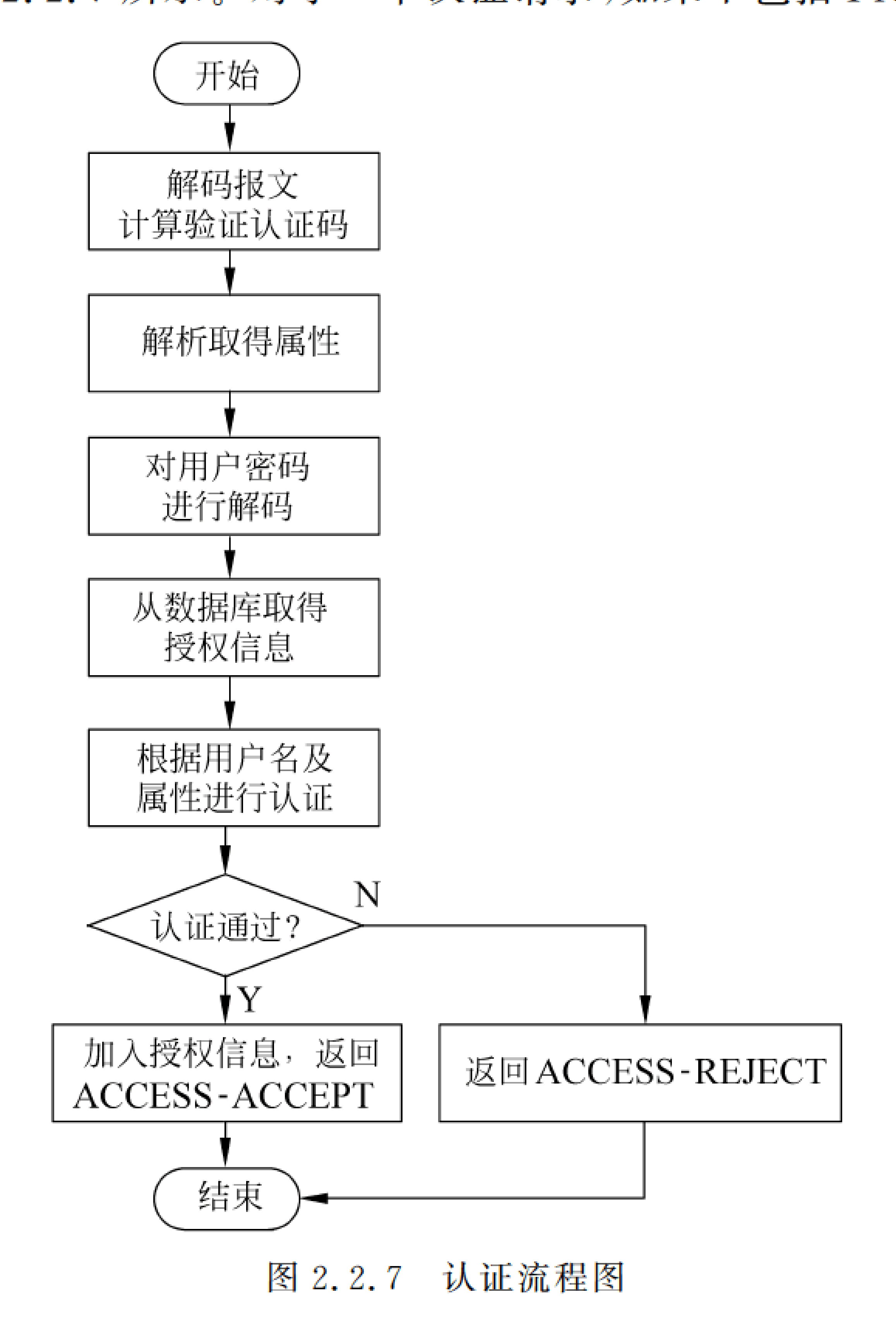


图 2.2.7 认证流程图

下要经过授权和认证两个过程。授权是从外部(文件或者数据库)获得一个用户信息的处理过程,以及检查这些信息是否能够对这个用户的验证。数据库以及文件等模块都属于授权模块。

认证方法在授权处理的过程中决定,因为一个特定的用户也许不能采用某种认证方法,所以在授权处理的过程中决定某用户采用哪种认证方法或者发送拒绝接入信息。

在一个认证和授权的处理过程中,有 3 个相关的队列:request 队列、config 队列和 reply 队列,每个认证请求数据包的属性都被填入 request 队列,认证和授权模块都可以将属性添加到 reply 队列中。这些被添加到 reply 中的属性将被收发包处理模块打包发送给客户端。

在授权处理开始的时候,系统为一个请求创建一个 request 属性队列和一个空的 config 属性队列。授权模块根据请求队列项中的属性(比如 User-Name)作为主键查询以及获取数据库中的所有相关记录,它查询 3 种类型的属性:验证属性、配置属性以及应答属性。它将取出的验证属性和 request 队列中传送过来的属性值相比较。如果数据库中根据主键取出的属性和 request 队列中的属性没有一个匹配,那么授权处理失败。如果有一个相匹配,那么这个匹配的属性要被加入到 config 队列中,同时所取出的应答属性都要被加入到 reply 队列中。授权模块最少要给认证模块传送一个属性,那就是 Auth-Type,这个属性将决定采用什么模块认证该用户。同时,授权模块还可以传送诸如用户密码或 hash 处理后的密码,以及登录限制等信息。

对于一个用户账号,我们只允许该用户名在同一时间内只能有一个计费服务的会话连接。也就是说,在用户享受我们提供的服务时,我们不让该用户名再次登录,这样才能保证我们的计费系统正常地实时计费。

2.2.6.2 改进 RADIUS 认证协议的安全

在目前 RADIUS 协议中,对用户密码属性采取的算法为 User-Password=Password (不足 16 位填 0) XOR MD5 (公用密钥+请求认证)。在这种算法中,破坏者可以截获数据报获得用户的密码,利用大量截获的数据进行分析,猜测用户密码,因而存在着潜在的网络安全问题。如果人为地从内部网络进行破坏,共享密钥的泄露会导致严重的灾难。而且用户的密码是以明文的方式存在于数据库中,数据库的管理员可以毫不费力地看到用户的明文密码,这对用户来说也是一个很大的安全隐患。

在本节的系统里,对目前的 RADIUS 协议的认证部分进行了一定的改进。改进后的 RADIUS 认证算法并没有放弃使用 MD5 算法,而是对 RADIUS 的认证方式作了一些改进,增加了网络的安全性能,使对 MD5 的攻击并不是轻而易举的;而在网络的传输过程中,对用户密码的保护也采取了适当的算法,增加用户认证的安全性。我们的 RADIUS 认证算法在 RADIUS 的客户端对用户密码生成 MD5 散列值,然后再进行相应的密码处理,而服务器数据库里保存的也是用户密码的 MD5 散列值。这样,网上破坏者最多也就是获得用户密码的 MD5 散列值,数据库管理员看到的也是用户密码的 MD5 散列值,从一定程度上增加了用户认证的安全性。

2.2.7　RADIUS 计费服务器

计费服务器需要满足以下几个要求：

（1）接收并处理标准 RADIUS 计费数据包；

（2）根据给定策略，能够准确并实时地计算当前某用户会话的费用，计算的最小时间粒度为分；

（3）对每项服务中用户的账户状态实时监控，并在需要时反馈到密钥管理服务器，保证用户不会恶意透支。

计费服务器的接收数据部分与认证/授权部分类似，同样地，计费服务器的主要任务是接收 RADIUS 计费数据包，根据包中的计费信息进行服务费用计算，所以采用线程池来进行计费数据包的处理和费用计算。由于支持实时计费，即需要对用户账户状态进行实时监控，所以需要有一个线程来做定时扫描和监控工作，保证整个计费过程的正确性，以及防止用户的恶意透支。

计费系统每隔一段时间产生定时消息，判断现有用户是否符合监控标准，如果符合，则启动监控线程对该用户进行监控。当被监视的用户余额不足时，停止该用户的服务。该监控线程还监控成员管理服务器发来的用户异常消息，如果客户端长时间没有响应，即视为用户已经退出服务，停止该用户的服务和计费。

此外，考虑到万一服务器出现问题而导致关闭或重启情况的发生，那么在服务器关闭或重启时应该对客户端会话进行处理，中断客户端的当前连接并停止计费。客户端需要重新进行身份认证和发送计费消息才可以享受服务。

2.2.8　系统冗余容错处理

系统的某些部件如果发生错误将会引起系统失效，这些错误包括存储器、网络连接设备、软件等，而认证/计费服务器属于重要的分布式系统，我们需要系统在部分部件出错时仍能够正常工作。由于系统中部件数量较多，系统可靠性显得尤为重要。

1. 冗余设计

解决容错的办法通常是使用冗余技术。冗余技术包括 3 类技术：信息冗余、时间冗余和物理冗余。物理冗余是指使用额外的部件使得整个系统能够容许一些部件的损失或失效。在组织冗余服务器时，有两种可用的办法：主动复制和主机后备。主机后备的基本思想是在任何时刻都有一台服务器是主机，它完成所有的工作，若这个服务器失效了，后备服务器将承担其任务。图 2.2.8 为主机后备示意图。

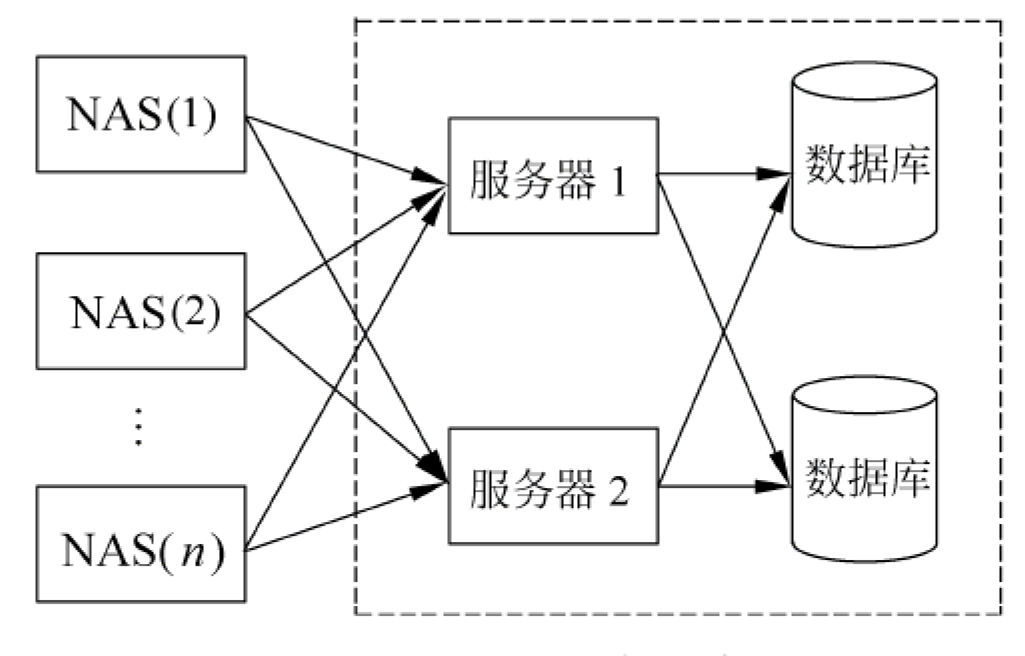


图 2.2.8　主机后备示意图

对于服务器的客户端，服务器 1 和服务器 2 是相同的，两个服务器并行工作，当其中一个服务器长时间没有响应时，即认为其失效，

此时向另一个服务器请求。这样,既保证容错又可以使服务器负载均衡。服务器与数据库采用了“一读两写”的方式,即对于查询操作,服务器向任意一个数据库查询均可;对于修改操作,如两数据库均工作正常,则服务器向两数据库同时发送命令,在两数据库都响应后,才认为操作成功。这种方式可以有效地保证数据库内容一致。

2. 异常处理设计

服务器的任何异常均被记录在日志文件里,并立即警告管理员,以便管理员进行相应检查。当数据库发生异常时,系统根据错误码判断是否为严重错误,如果是,则关闭相应数据库连接,并采用另一数据库继续进行认证、计费服务,同时发出警告,提示管理员检查相应数据库。管理员处理故障后,可在服务器正常工作的情况下重新连接并恢复数据库。

2.3 下一代 AAA 协议——Diameter 协议

随着新接入技术的引入(如无线接入、DSL、移动 IP 和以太网)和接入网络的快速扩容,越来越复杂的路由器和接入服务器大量投入使用,对 AAA 协议提出了新的要求,使得传统的 RADIUS 结构的缺点日益明显。目前,3G 网络正逐步向全 IP 网络演进,不仅在核心网络使用支持 IP 的网络实体,在接入网络也使用基于 IP 的技术,而且移动终端也成为可激活的 IP 客户端。如在 WCDMA 当前规划的 R6 版本就新增了以下特性:UTRAN 和 CN 传输增强;无线接口增强;多媒体广播和多播服务(multimedia broadcast multicast service,MBMS);数字权限管理(digital rights management,DRM);WLAN-UMTS 互通;优先业务;通用用户信息(GUP);网络共享;不同网络间的互通等。在这样的网络中,移动 IP 将被广泛使用。支持移动 IP 的终端可以在注册的家乡网络中移动,或漫游到其他运营商的网络。当终端要接入到网络,并使用运营商提供的各项业务时,就需要严格的 AAA 过程。AAA 服务器要对移动终端进行认证,授权允许用户使用的业务,并收集用户使用资源的情况,以产生计费信息。这就需要采用新一代的 AAA 协议——Diameter 协议。此外,在 IEEE 的无线局域网协议 802.16e 的建议草案中,网络参考模型里也包含了认证和授权服务器,以支持移动台在不同基站之间的切换。可见,在未来移动通信系统中,AAA 服务器占据了很重要的位置。

IETF 的 AAA 工作组已同意将 Diameter 协议作为下一代的 AAA 协议标准。Diameter 协议的设计目的是创建一个能够充分满足目前乃至今后 IP 网络(包括 NGN 和 3G 等)用户访问控制要求的 AAA 协议。Diameter 协议在设计时,克服了现有 AAA 技术的许多不足,并保持了对广为使用的 RADIUS 的兼容,而且它被设计得非常灵活,容易进行新应用的扩展,以满足新的要求。所以它不但为互联网所采用,也受到了下一代移动通信网的欢迎。在 ITU,3GPP 和 3GPP2 等国际标准组织中,都已经正式地将 Diameter 协议作为未来通信网络的首选 AAA 协议。目前 Diameter 协议族中部分协议已经当作标准得以发布,部分协议还只是 Internet 技术草案,还有很多问题有待解决。

2.3.1　Diameter 协议概述

Diameter 协议包括基础协议(base protocol)[12]、网络接入服务(Network Access Server, NAS)协议[13]、扩展认证协议(extensible authentication protocol, EAP)[14]、移动 IP(Mobile IP, MIP)协议[15]、密码消息语法(cryptographic message syntax, CMS)协议[16]、信任控制(credit-control)[17]协议。Diameter 协议支持移动 IP、NAS 请求和移动代理的认证、授权和计费工作,协议的实现和 RADIUS 类似,也是采用 AVP(attribute value pair, 属性值对)来实现,但是其中详细规定了错误处理机制,采用 TCP 协议,支持分布式计费,克服了 RADIUS 的许多缺点,是最适合未来移动通信系统的 AAA 协议。

2.3.1.1　Diameter 协议的体系结构

Diameter 协议族包括基础协议和各种应用协议。Diameter 协议层次结构如图 2.3.1 所示。

<table>
<tr><td>移动应用协议</td><td>网络接入协议</td><td>EAP 应用协议</td><td>多媒体应用</td><td>其他应用</td></tr>
<tr><td colspan="3">基础协议</td><td colspan="2">CMS 协议</td></tr>
<tr><td colspan="5">TLS</td></tr>
<tr><td colspan="2">TCP</td><td colspan="3">SCTP</td></tr>
<tr><td colspan="5">IP/IPSec</td></tr>
</table>

图 2.3.1　Diameter 协议层次结构

基础协议提供了作为一个 AAA 协议的最低需求,是 Diameter 网络节点都必须实现的功能,包括节点间能力的协商、Diameter 消息的接收及转发、计费信息的实时传输等。协议元素由众多命令和 AVP 构成,可以在客户机、代理、服务器之间传递认证、授权和计费信息。命令代码、AVP 值和种类都可以按应用需要和规则进行扩展。基础协议可以作为一个计费协议单独使用,但一般情况下需要与某个应用一起使用。

应用协议则充分利用基础协议提供的消息传送机制,规范相关节点的功能以及其特有的消息内容,来实现应用业务的 AAA。

Diameter 的网络接入服务(NAS)用于实现网络接入环境下的 AAA 服务。由 NAS 处理用户的接入请求,将收到的用户认证信息转送给 Diameter 服务器;服务器对用户进行认证授权,将结果发给 NAS;NAS 将结果发回给用户,并根据结果进行相应处理。

Diameter 密码消息语法(CMS)协议实现了协议数据的 Peer-to-Peer(端到端)加密。由于 Diameter 网络中存在不可信的中继(relay)和代理(proxy),而 IPSec 和 TLS(transport layer security protocol,安全传输层协议)又只能实现跳到跳的安全,所以 IETF 定义了 Diameter CMS 应用协议来保证数据安全。

Diameter 可扩展认证(EAP)协议提供了一个支持各种认证方法的标准机制。Diameter MIP 应用协议允许用户漫游到外部域,并在经过鉴权后接收外部域服务器和代理提供的服务。

2.3.1.2 Diameter 协议的工作原理

在 Diameter 协议中,包括多种类型的 Diameter 节点。除了 Diameter 客户端和 Diameter 服务器外,还有 Diameter 中继、Diameter 代理、Diameter 重定向器和 Diameter 协议转换器等。

为处理用户的接入,Diameter 客户端通过 Diameter 基础协议和应用协议,与 Diameter 服务器进行一系列的信息交换,而这样一个从发起到中止的一系列信息交互,在 Diameter 协议里被称为一个用户会话(user session)。一个用户会话的建立,一般由 Diameter 客户端发起,中间可以途径若干 Diameter 代理、重定向器或协议转换器,一直延伸到 Diameter 服务器。

一般的 AAA 业务可以大致分成两类:一类包括用户的认证和授权,可能还包括计费(如移动电话业务);另一类则是仅包括对用户的计费(如目前的主叫拨号接入业务)。为此,Diameter 基础协议提供对应的两类用户会话,为上层的应用服务。

当用户被允许接入时,Diameter 客户端将根据情况产生针对用户的计费信息。这些计费信息将被封装在具体 Diameter 应用专有的 AVP 内,由 Diameter 基础协议中定义的计费请求(accounting request,ACR)消息,传送给 Diameter 服务器。服务器将响应计费应答(accounting answer,ACA)消息,指示计费成功或拒绝。客户端只有在收到成功的计费响应时,才能清除已经被发送的计费记录。当收到计费拒绝指示时,客户端将中止用户接入。Diameter 支持实时计费,客户端通过在首次计费请求/响应交互过程中协商好的计费消息间歇时间,定时地向服务器发送已收集的计费信息。这种实时计费确保了对用户费用的实时检查。

Diameter 客户端必须支持 IPSec,可以支持 TLS;而 Diameter 服务器必须支持 IPSec 和 TLS。IPSec 主要应用在网络的边缘和域内的流量,而域间的流量主要通过 TLS 来保证安全。

2.3.2 Diameter 协议格式

1. Diameter **协议的报文格式**

一个基本 Diameter 的消息包由一个包头和多个 AVP 组成,包头的结构如图 2.3.2 所示,其说明如下。

第 1 字节	第 2 字节	第 3 字节	第 4 字节
版本号	消息长度		
消息标志位	命令代码		
Vendor-ID			
点到点标识			
端到端标识			
AVPs...			

图 2.3.2 Diameter 协议报文格式

版本号：必须为 1，以标明 Diameter 的版本。

消息长度：指明这个 Diameter 消息包的长度。

消息标志位：这个 8 位的字段其中一位标识这个包是请求还是回复，另一位表明是本地消息还是转发的代理消息，还有一位标识是否是出错消息，其余各位保留。

命令代码：每一个 Diameter 消息都必须在包头中有一个命令代码，用来指定对消息所作的操作。

Vendor-ID：运行商的 ID。

点到点标识：一个无符号的 32 位字段，用于匹配请求和相应的回复。

端到端标识：一个无符号的 32 位字段，用于发现重复的消息，防止重放攻击。

Diameter 中还采用了 AVP 来携带更多的 AAA 信息，Diameter 的 AVP 用来携带认证授权和计费信息，以及路由、安全信息等。

2. Diameter 协议的 AVP 格式

AVP 的格式如图 2.3.3 所示。AVP 代码和 Vendor ID(厂商标识)一起来保证这个属性的唯一性，用于和其他属性相区分，AVP Flag 用来标示 AVP 的某些性质，数据部分携带详细的信息。

第 1 字节	第 2 字节	第 3 字节	第 4 字节
AVP 代码			
AVP 标志位	AVP 长度		
Vendor-ID (opt)			
数据…			

图 2.3.3　Diameter 协议中 AVP 的格式

AVP 的格式中头 4 个字节是 AVP 代码，下面 4 个字节由 8 比特的 AVP 标志和 24 比特的 AVP 长度(包括 AVP 头部长度)构成。AVP 标志用于通知接收端如何处理这个 AVP 以及该 AVP 是否包含厂商标识。在长度之后是 4 个字节的厂商标识(可选)，用于标识特定厂商自定义的 AVP。厂商标识后的字节是 AVP 的数据部分。AVP 的数据类型，目前包括字符串、32 比特整数、64 比特整数、32 比特浮点数、64 比特浮点数以及 AVP 组等。

2.3.3　Diameter 与 RADIUS 的比较

目前，RADIUS 已经是网络环境中的 AAA 标准。尽管有些产品针对 RADIUS 进行了改进，性能有所提高，但是由于 RADIUS 存在着先天缺陷，限制了其推广与应用。而 Diameter 吸取了 RADIUS 部署和运作中的各种教训，从设计上就克服了 RADIUS 功能上的限制，并将最终代替 RADIUS。自从 Diameter 协议成为 IETF 草案以后，IETF 就组织志愿者开发 Diameter 服务器和客户端。而且，它的源代码作为开放软件予以公布及讨论，其软件包命名为 opendiameter，用来提供 Diameter 协议的实现原型和 C++ API。

我们从以下几个方面对 Diameter 和 RADIUS 协议进行比较。

（1）安全性

RADIUS 协议存在着明显的安全弱点，这些安全问题主要来自于 RADIUS 的设计和实施，如用户密码保护技术的缺陷、认证值的使用不当等。由于 RADIUS 的安全是基于安全密钥的，这样在认证或计费需要通过代理链的情况下就无法提供端到端的安全性。

RADIUS 协议并不要求支持 IPSec，而 Diameter 基础协议规定 Diameter 客户端必须支持 IPSec，可以支持 TLS，Diameter 服务器必须支持 IPSec 和 TLS，在没有任何安全机制（TLS 或 IPSec）的情况下不能使用 Diameter。这样就提供了统一的传输层面上的安全，使得域内和域间 AAA 的部署成为可能。

（2）扩容

在 RADIUS 协议中没有中继器和重定向器，这样，当 NAS 需要支持更多的用户时，就需要增加新的 AAA 服务器。而 Diameter 能够很好地支持中继、代理和重定向器，这样就可以把用户分组，把系统管理的能力分发到每个组，也能对来自不同用户组的请求加以集合，并转发到合适的目标，同时还能很好地实现负载均衡。

（3）传输可靠性

RADIUS 运行在 UDP 之上，尽管可以在应用层实现重传，但仍很难有效地处理代理链情况下的可靠传输，更何况这样也会使不同实现的可靠性有所不同，增加互操作的难度，而传输的不可靠性会对计费产生影响。相反地，Diameter 运行在可靠的传输协议之上，如 TCP 和 SCTP(stream control transmission protocol，流控制传输协议)。

（4）漫游支持

由于 RADIUS 协议中并没有详细定义对代理的支持，并且缺乏可审计性及传输层面上的安全等，使得基于 RADIUS 的漫游不仅易于受到外部的攻击，还会受到其他漫游用户欺骗行为的影响。因此，RADIUS 并不适合应用于大规模漫游环境中。而 Diameter 协议中明确定义了域间漫游、消息的路由并提供强大的安全性，因此，Diameter 能提供安全的漫游，且易于升级。

（5）故障切换

RADIUS 中没有明确定义故障转移(failover)和故障恢复(failback)机制，尤其是在代理链情况下的故障问题根本没有提及，这就会对 RADIUS 的实现和不同产品间的互操作性产生很大影响。Diameter 支持应用层的确认机制，定义了故障切换的算法和状态机。协议中定义的监视消息使应用能够快速检测到传输层或应用层的故障，从一个发生故障的对等端切换到其他对等端。

（6）扩展性

Diameter 非常易于扩展，这主要得益于它将基础协议和支持各种业务的应用协议分开的设计思想。例如，目前的 Diameter 不支持批量计费，如果需要的话，只需在基础协议上扩展一个应用就可以了。

24 AAA 在无线网络中的应用

21 世纪网络的发展进入了一个新的阶段，特别是在 20 世纪的最后 10 年，移动通信，尤其是数字移动通信的发展之快和应用之广，大大超出了人们的预料和专家们的预测。随着

信息交流的增加，无论是工作还是学习、娱乐，人们需要在实时地了解和掌控信息的同时不影响正常的工作和生活。这就要求 Internet 适应新的需求，发展新的技术。而笔记本电脑、数字手机、PDA 等通信技术的发展，给 Internet 和移动通信的结合提供了硬件基础。

在通过有线网络访问接入 Internet 的时代，多台计算机之间交换信息是用一系列复杂的规则，即网络协议来实现的。但在制定这些协议时，几乎所有计算机都是不需要移动的。所以，大多数目前存在的通信协议都不足以应付快速移动中的计算机通信。

移动 IP 是一种在全球 Internet 上提供移动功能的方案，它具有可扩展性、可靠性和安全性，并使节点在切换链路时仍然可以保持正在进行的通信。值得注意的是，移动 IP 提供了一种 IP 路由机制，使移动节点可以以一个永久的 IP 地址连接到任何链路上。但另一方面，移动 IP 通信既经过了无线链路，又通过了有线链路，增加了受外来攻击的概率。因此人们在享受丰富的信息资源和便捷服务的同时，也面临着更大的网络信息安全方面的威胁。

移动 IP 业务的使用需要 Internet 提供支持移动 IP 的 AAA 服务，即移动用户的认证、授权和计费服务。当移动节点(mobile node，MN)移动到外地网络时，MN 需要对外地代理或接入设备进行认证，以确定对方的有效性，外地代理也需要对 MN 进行身份认证，以防止非法用户的攻击行为。授权和计费主要涉及 MN 在外地网络上的资源的使用权和使用情况。

2.4.1　基本模型

AAA 本身就支持跨域通信的管理模式，其基本模型如图 2.4.1 所示[18]。在此模型中，用户向服务员提出服务请求，服务员则要求用户提供“安全证书”，以验证用户是否可以使用资源。同时服务员将此信息交给 AAA 服务器，AAA 服务器处理从服务员发来的 AAA 请求。同时，本地域的信息可能不足以对此用户进行 AAA 计算，但应该配置了足够的信息以向用户的家乡(或归属)AAA 服务器(AAAH)建立安全关联，向其提交用户的安全证书；AAAH 在对用户进行验证后返回用户授权信息。本地 AAA 服务器(AAAL)将根据此授权信息通知服务员向用户提供相应服务。

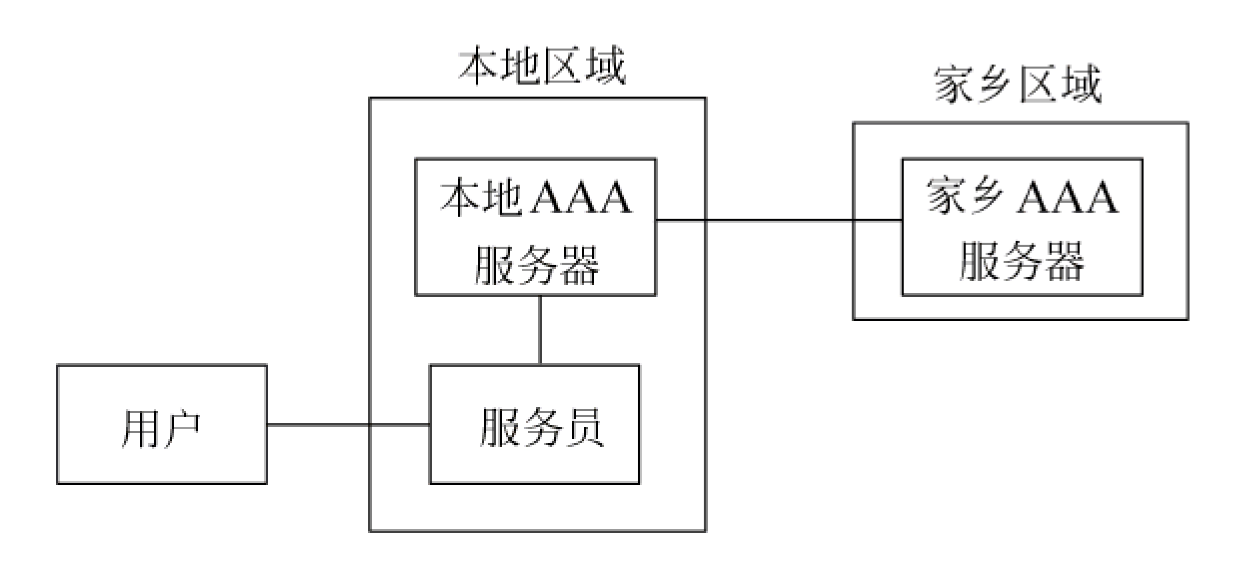


图 2.4.1　AAA 服务的基本模型

2.4.2　AAA 协议漫游的需求

(1) 支持可靠的 AAA 传输机制。主要包括：路由各跳之间必须有一种重传和失败恢复机制；传输机制必须可以向 AAA 应用显示信息已传到下一个对点的 AAA 应用中或者显

示超时;重传由可靠的 AAA 传输机制控制,而不是由低层协议(如 TCP)控制;确认工作应该可以由 AAA 消息来承担;AAA 响应应该及时,以免超时和重传。

(2) 在 AAA 消息中传输数字证书,目的是最小化与 AAA 交互的往返次数。

(3) 在各跳(AAA 节点)之间提供消息完整性和身份验证。

(4) 对于所有的授权和记录数据支持重放和抗抵赖的功能。AAA 协议必须要求记录数据与前一个认证消息匹配。

(5) 通过双向配置和 broker(中继代理)AAA 服务器,实现提供服务处和本地网络之间的交互记录和调节工作。

2.4.3 移动 IP 的 AAA

使用移动 IP 的客户除了需要 AAA 服务器提供基本的功能要求和 IP 连接的要求之外,还需要 AAA 服务器提供特殊的服务。为了理解移动 IP 模型的应用,我们将移动节点作为用户,而将外来代理充当访问服务器的角色。移动 IP 的 AAA 服务模型如图 2.4.2 所示。

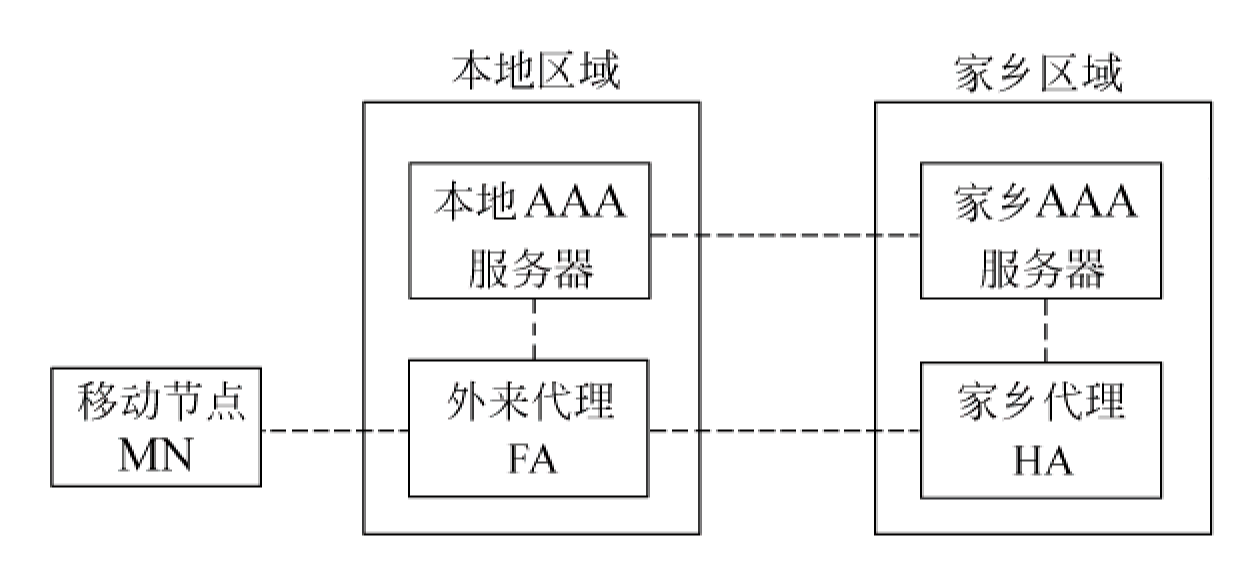


图 2.4.2 Mobile IP 的 AAA 服务模型

移动节点 MN 向外来代理 FA 提出带有安全证书的 Mobile IP 注册请求,FA 将此请求交给本地服务器 AAAL。AAAL 通过安全关联将请求转发至用户的归属 AAA 服务器 AAAH。AAAH 对用户进行认证并授权完成之后向用户家乡代理(home agent,HA)发送包含用户授权信息的注册请求,HA 处理此请求并产生响应。然后 AAAH、AAAL 和 FA 依次响应,MN 完成 Mobile IP 注册和 AAA 服务的验证授权。

目前实现了 AAA 对 Mobile IP 诸多支持的协议为 Diameter 协议。Diameter 移动 IP 应用使 Diameter 服务器能够对移动 IP 业务进行认证、授权,并收集计费信息,使网络运营商可以有效地控制对移动节点的接入。由于 Diameter 基础协议提供了域间交互的能力,所以漫游的移动节点也能从外地业务提供商处获得接入业务。

随着无线网络协议的日益成熟,为移动用户提供流媒体服务成为可能。我们有必要对流媒体系统中引入 Mobile IP 后的认证结构进行一些研究和探讨。流媒体系统中 Mobile IP 的引入将对流媒体数据下发、编码以及认证方式都产生较大的影响。

引入 Mobile IP 的流媒体平台结构如图 2.4.3 所示。该系统支持客户端移动或固定节点,对于固定节点来说,其认证计费流程未发生改变,在申请服务时提交用户认证请求,服务过程中与流媒体服务器交互计费信息。对于移动节点来说,认证包含两个含义:通信节点的认证和使用此节点的用户认证。在其服务启动时首先需要通过代理来确定节点所处的位

置，然后向流媒体系统提交服务请求并对用户信息进行认证；在节点移动过程中，移动节点需要向家乡代理注册，为阻止拒绝服务攻击，注册消息要求进行认证，这是对通信节点的认证。

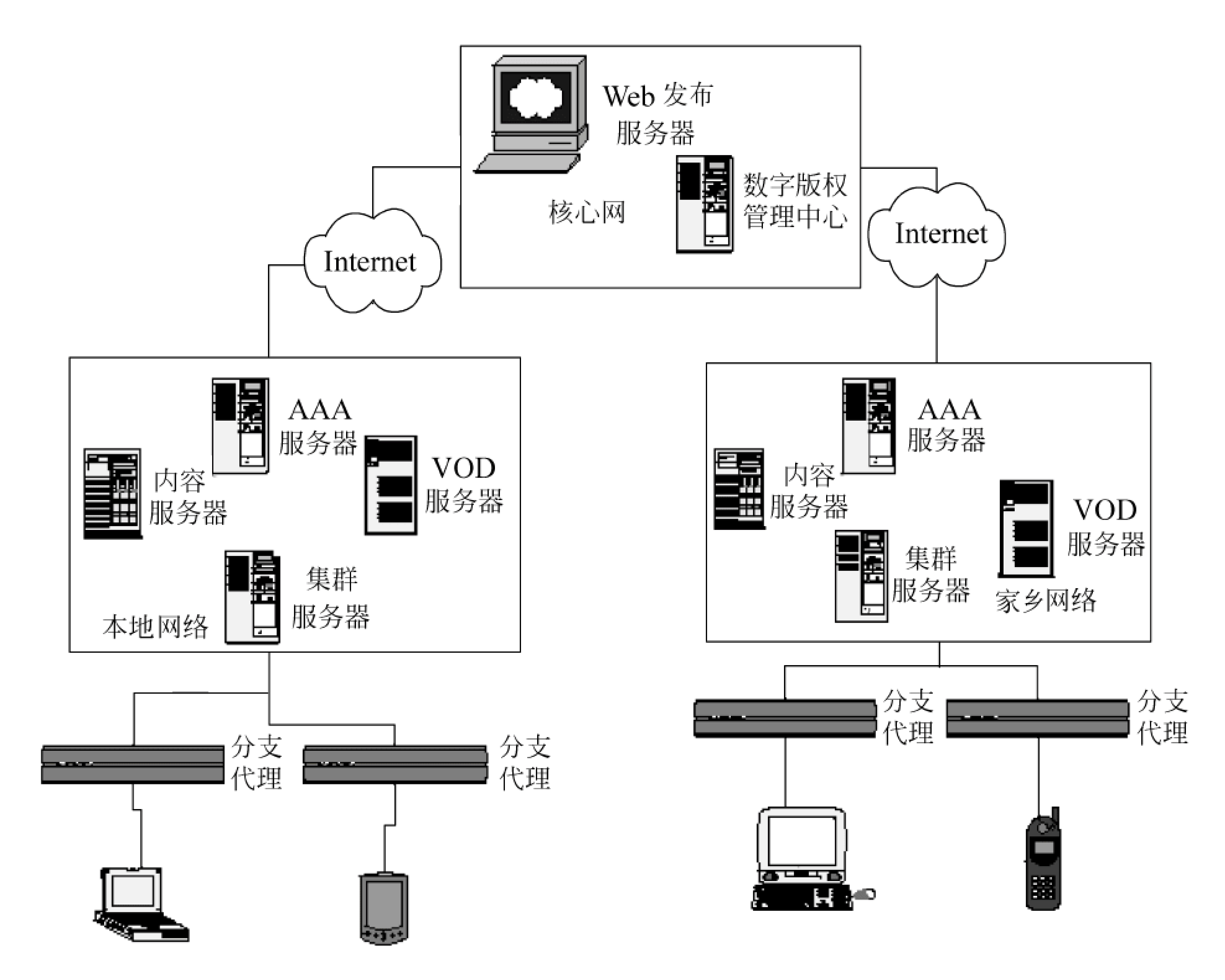


图 2.4.3　流媒体系统中 Mobile IP 架构图

2.4.4　3G-WLAN 互联中的 AAA

目前提供宽带无线数据通信的主要方式为 WLAN 和 3G。3G 作为一个完整的移动通信系统可以为用户提供无所不在的连接性，并且有成熟的漫游协议。但是 3G 的投资规模相当庞大，数据峰值传输速率也只有 2 Mbps 左右。另一方面，WLAN 能够提供远超过 3G 的带宽，但只能适用于公司、旅馆、机场等所谓的“热点”地区，而且不同 WLAN 业务提供商之间的网络没有漫游协议，同时缺乏充分的安全措施和完整的结构。3G 系统的优势在于计费管理、漫游与安全性；WLAN 系统的优势在于高带宽和低投资成本。通过 3G 与 WLAN 的互联可以实现优势互补。因此，第三代移动通信合作计划(3GPP)开始研究 3G 与 WLAN 的互联，并确定互联的基本原则：3G 与 WLAN 互联必须尽量减少对 WLAN 以及 3G 标准的影响，即保持 WLAN 标准不变，同时使得对 3GPP 现存规范的修改最小化。两者互联的一个挑战就是如何协调和加强网络的安全体系[19]。

按照互联的级别，ETSI BRAN 在 ETSI TR 101 957[20] 制定了“紧互联”和“松互联”两种完全不同的方案[21,22]。3GPP 目前正在制定的标准化工作是 WLAN 与 3G 互联的可行性和体系结构[23,24]。在当前的标准化版本 R6 中，3GPP 与 WLAN 的互联采用了“松互联”

的方案。

3GPP-WLAN 互联结构在 3GPP TS 23.234[23]中定义，该文档中给出了非漫游情况和漫游情况下的参考模型。在非漫游情况下，WLAN 接入网关(WLAN access gateway，WAG)和分组数据网关(packet data gateway，PDG)位于归属公众陆地移动通信网(home public land mobile network，HPLMN)中，如图 2.4.4 所示。当用户设备接入 WLAN 网络时，首先需要进行身份认证。3GPP AAA 服务器从归属用户服务器(home subscriber server，HSS)/归属位置登记器(home location register，HLR)获得认证矢量，通过 WLAN 接入网执行认证和密钥协商(authentication and key agreement，AKA)过程。一旦认证通过，用户获得接入权限，WLAN 接入网就可以保证用户设备接入 IP 网。如果 IP 网络为 Internet/Intranet，用户数据直接从 WLAN 接入网路由到 Internet/Intranet。当用户使用 3GPP 网络提供的业务时，数据从 WLAN 接入网通过 WAG 和 PDG 路由到 3GPP 网络的实体，从而获得所需的业务。接入网络后由计费系统进行计费。

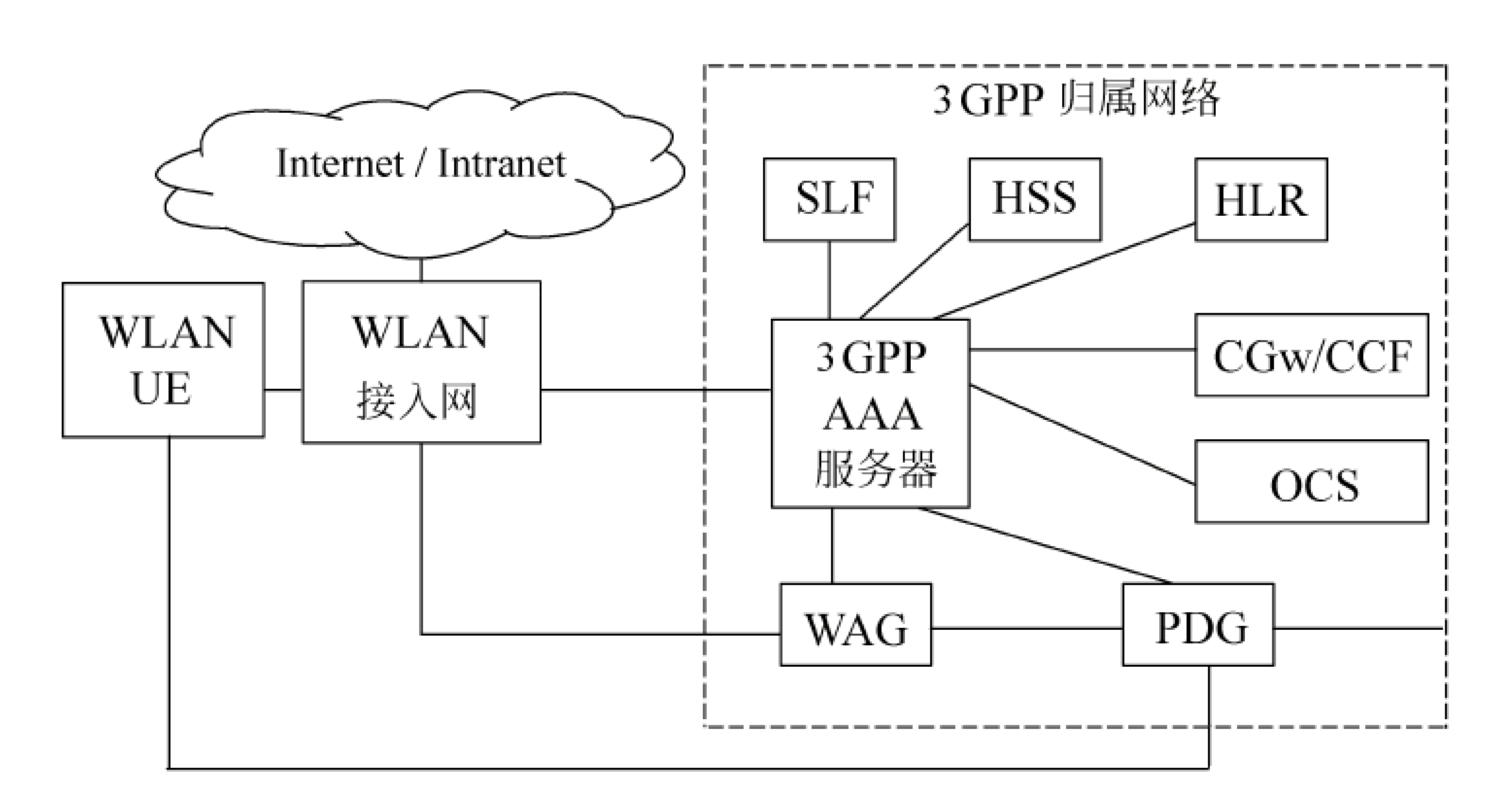
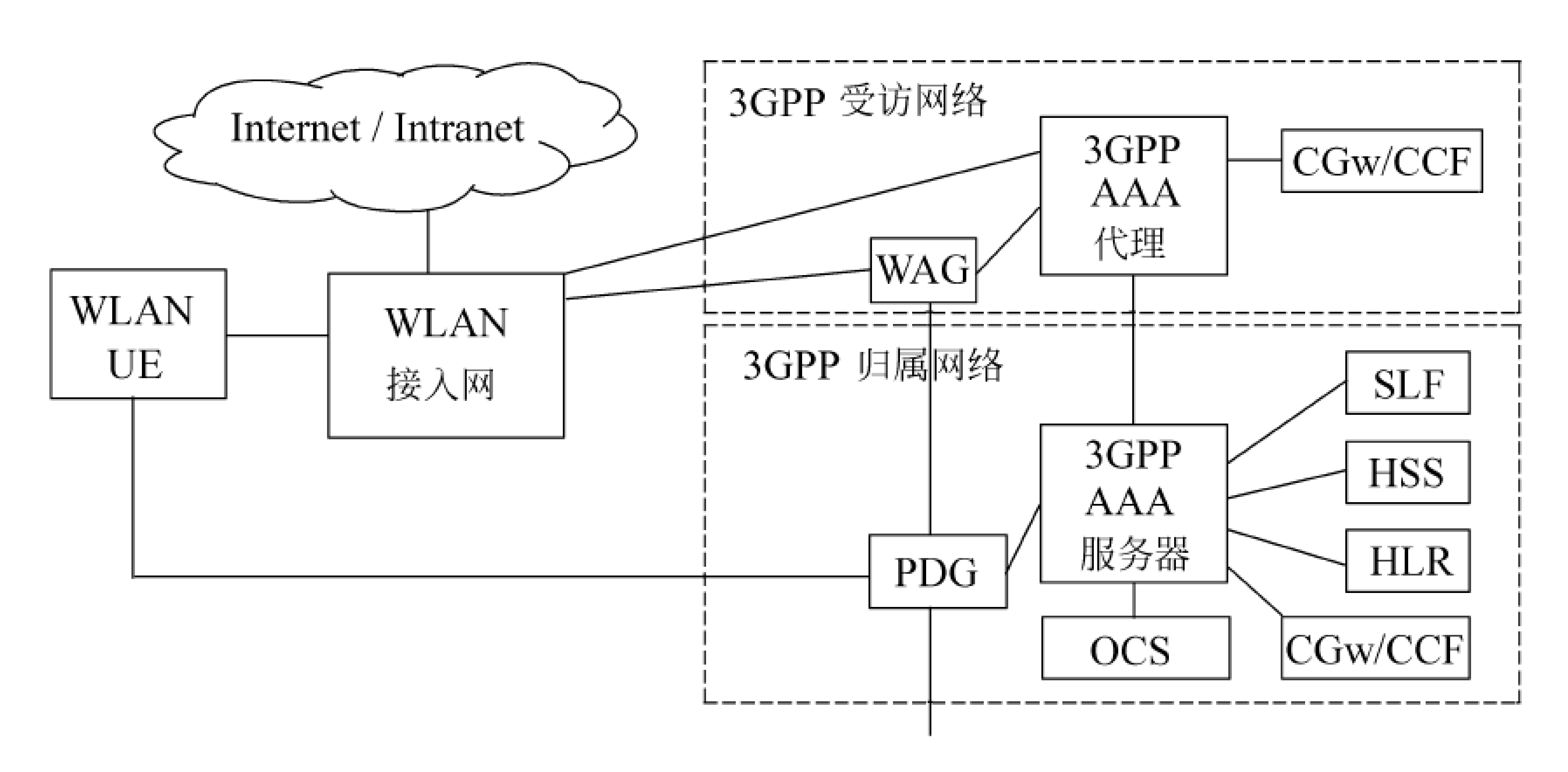


图 2.4.4 非漫游情况下的互联参考模型

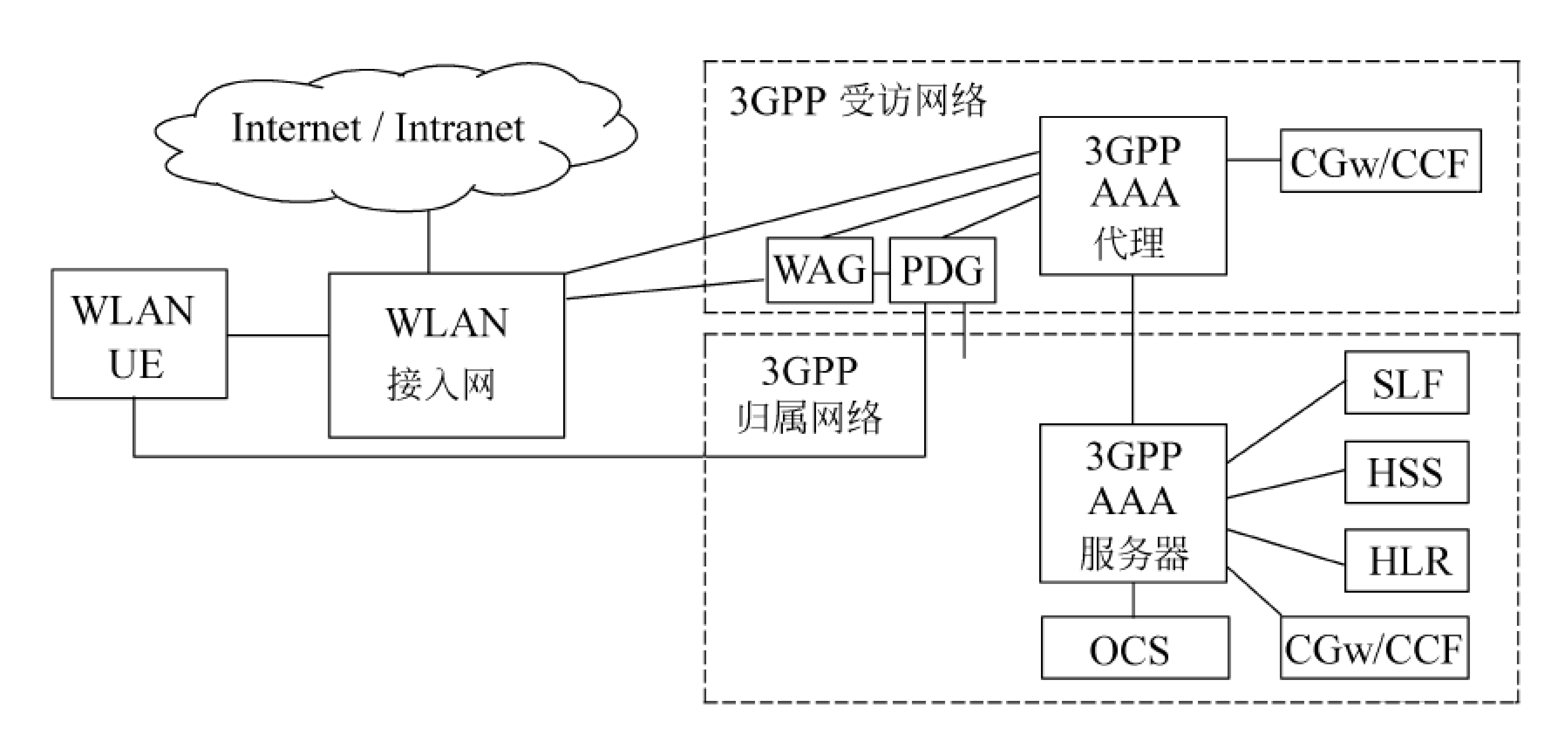
在漫游情况下，WAG 应该位于受访公众陆地移动通信网(visited public land mobile network，VPLMN)中，如图 2.4.5 所示。PDG 所在的位置与用户使用由谁提供的分组域业务有关，如果用户使用由 HPLMN 提供的 3GPP 分组域业务，则 PDG 位于 HPLMN 中，如图 2.4.5(a)所示；如果用户使用由 VPLMN 提供的 3GPP 分组域业务，则 PDG 位于 VPLMN 中，如图 2.4.5(b)所示。与漫游情况不同，用户需要选择可用和合适的受访网络，通过受访网络中的 3GPP AAA 代理与归属网络中的 3GPP AAA 服务器进行身份认证。其他过程与非漫游情况相同。

在互联网络中，首选的是 Diameter 协议。在 Diameter 协议中，所有的 AAA 数据都以属性值对(AVP)的方式发送[25]。AVP 可以任意加入 Diameter 消息中，以实现 AAA、性能协商和消息路由，通过增加新命令和 AVP 可以实现功能的扩展。

漫游情况下，使用 Diameter 协议的 AAA 协议的体系结构如图 2.4.6 所示[26]，非漫游情况下的体系结构缺少 WLAN AAA 代理和 3GPP AAA 代理两部分。需要注意的是，用来认证 UE 和 3G 归属网络的方法由 UE 和 UE 的归属网络的 3G AAA 服务器来执行，WLAN 只需支持 EAP 和 IEEE 802.1x 定义的 EAP-over-LAN(EAPOL)。



(a) PDG 位于 3GPP 归属网络



(b) PDG 位于 3GPP 拜访者网络

图 2.4.5　漫游情况下的互联参考模型

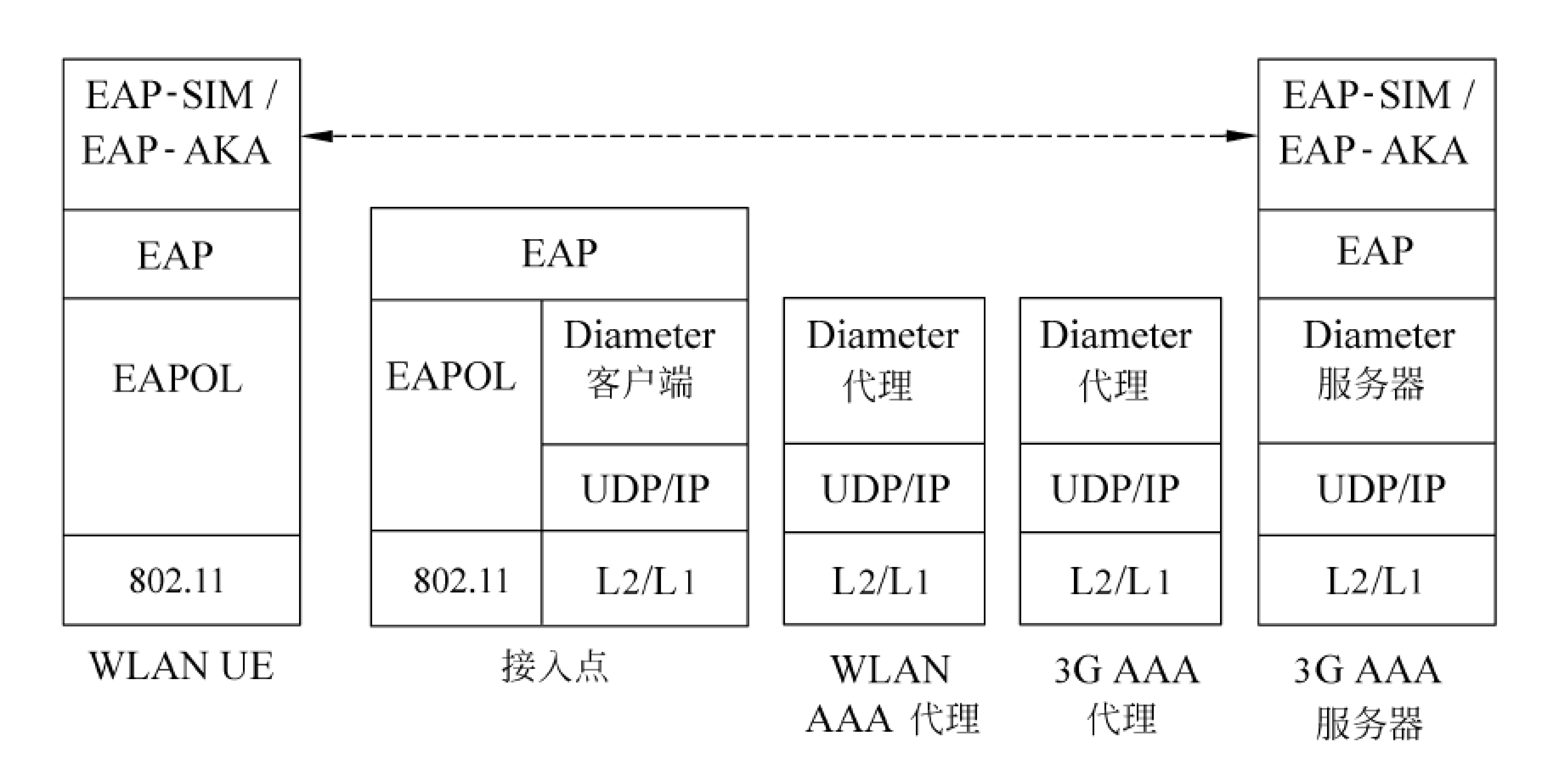


图 2.4.6　AAA 协议的体系结构

图 2.4.7 表示了 3G 与 WLAN 之间一次成功的 AKA 执行过程。3GPP 选择 EAP-AKA 协议来交换 AKA 认证信息。EAP 协议是一个通用协议，特点是在链路控制阶段没

有选定一种认证机制，而把这一步推迟到认证阶段。EAP 协议无需 IP，有自己的流控制机制，能够删除复制的报文和重传丢失的报文，可在不同链路层上使用。因为 UMTS AKA 支持相互认证和强大的密钥推导，因此 EAP-AKA 协议可以非常可靠地将 UMTS 认证机制封装到 EAP 中。

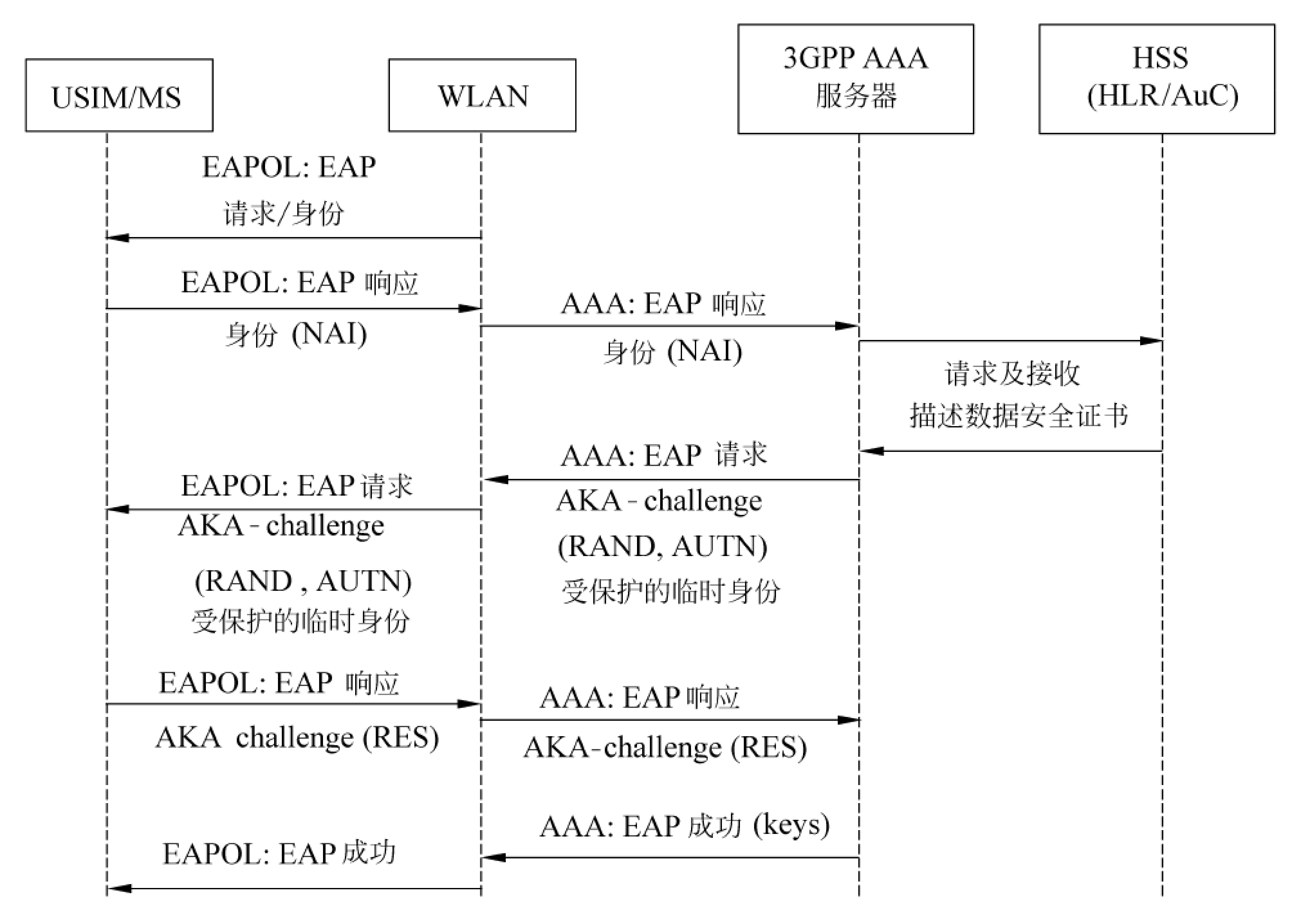


图 2.4.7 3G 与 WLAN 之间一次成功的 AKA 过程

在 3G-WLAN 互联中的 AKA 机制类似于 UMTS(universal mobile telecommunications system，通用移动通信系统)的 AKA 机制，而 UMTS AKA 机制从 GSM 的认证机制发展而来，其认证都是基于一个对称的密钥，该密钥保存在用户的 USIM 卡和相应归属网络的 HSS/HLR 中，采用基于“挑战-响应”的协议。AKA GSM 中的许多已知的弱点在 UMTS 中得到了修正，但还是有许多缺点影响到 EAP-AKA 机制[27]。

我们还可以使用许多如 EAP-TLS 和 EAP-TTLS(extensible authentication protocol-tunneled transport layer security，EAP-隧道传输层安全协议)等标准认证协议来增加安全性。EAP-TLS 和 EAP-TTLS 协议是经过修改了的安全套接层(secure socket layer，SSL)协议。EAP-TLS 协议基于 SSL 协议的 3.0 版本，使用 EAP 代替 TCP 来执行 SSL 的握手过程。该协议要求认证者和认证服务器都有一个证书。在互相认证的过程中，双方都需要使用该证书来认证身份和私钥。与 EAP-AKA 不同，EAP-TLS 采用基于公钥密码机制(PKC)[28]，因此，它不需要中心服务器来与移动台共享一个密钥，而且它还是可以升级的。EAP-TLS 提供的是一种在客户端和认证服务器之间互相认证的机制。

图 2.4.8 所示的是基于 EAP-TLS 的 AKA 机制。作为 EAP 请求的一部分，AAA 服务器一方面把自己的证书提供给客户端，同时要求客户端的证书。客户端首先认证服务器的证书，然后在 EAP 响应消息中包含自己的证书，同时开始协商加密方案(密码和加密算法)。

在通过客户端证书的认证后，服务器给出会话的加密方案。

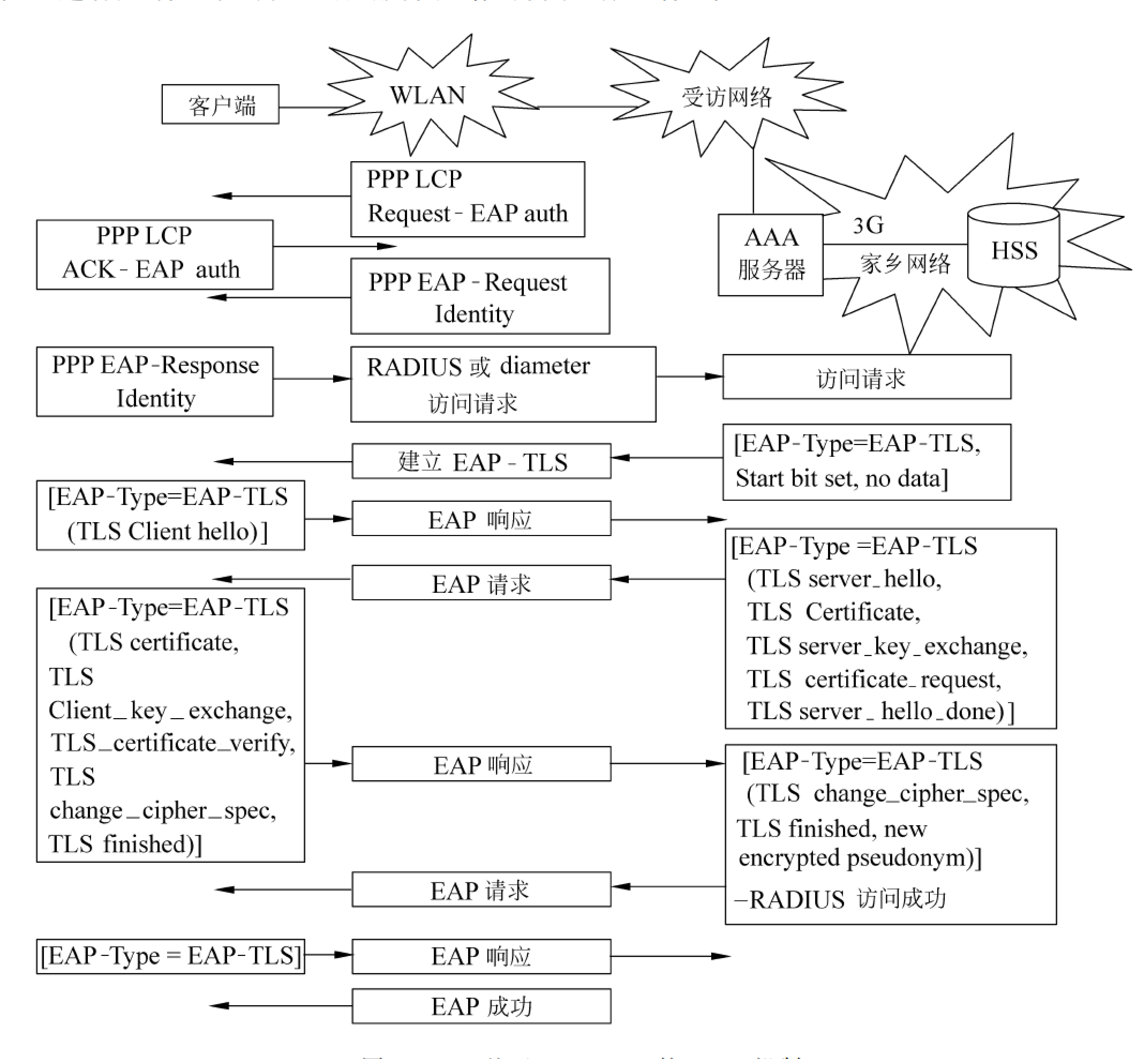


图 2.4.8 基于 EAP-TLS 的 AKA 机制

EAP-TTLS 协议是 EAP-TLS 的修订版本[29]。该协议通过使用安全连接来扩展认证的协商过程，该安全连接在客户端和服务器之间使用 TLS 握手来交换额外的信息。该安全连接允许服务器使用现有的、广泛应用的认证机制来认证客户端。客户端的认证协议可以是 EAP 或者其他认证协议，如“挑战-握手认证协议(CHAP)”。EAP-TTLS 能够快速地配置一个现有的、用户和服务器之间已经共享安全密钥的体系结构，允许使用遗留的基于密码的认证协议，同时能够保护遗留协议的安全免受窃听、“中间人”和其他密码攻击。

图 2.4.9 显示了一个典型的 EAP-TTLS 认证协议。其中，服务器的认证使用了 TLS 协议，客户端的认证使用了基于密码的认证协议 CHAP。客户端把 User-Name、CHAP-Challenge 和 CHAP-Password 属性对组(AVPs)通过隧道传输到服务器。在接收到客户端的属性对组后，服务器必须验证 CHAP-Challenge 属性和 CHAP-Password 属性对中的 CHAP 标识符是否等于挑战原来产生的值。如果其中任意一项不相等，那么服务器就拒绝客户端。否则，服务器在 Access-Request 请求中把属性对组提交到 AAA 服务器。

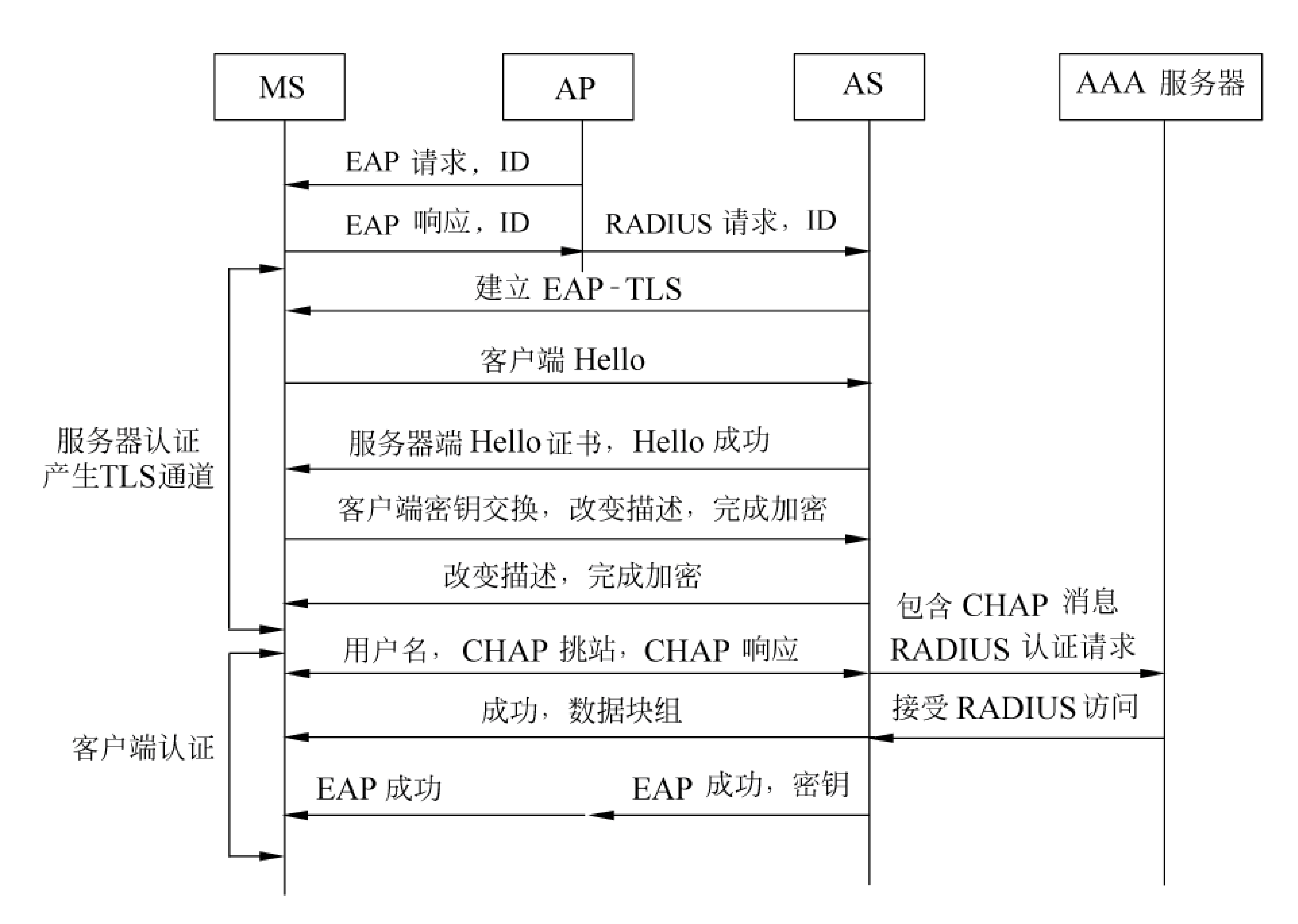


图 2.4.9　使用 CHAP 的 TTLS 协议

2.5　多级安全域的认证模型

2.5.1　多级安全域的格模型

在大型复杂的网络系统中，存在着一系列相互信任或不相互信任的安全自治网络域。特别是 Internet 的异构性和复杂性更体现了这一点。因此，有必要对安全域的关系模型进行分析。

在 ISO/IEC CD 10181—1 中，安全域被定义为：

A set of elements, a security policy, a security authority and a set of security relevant activities in which the set of elements are subject to the security policy, administered by the security authority, for the specified activities.

安全域是由一组在同一个管理器管理下的安全主体和客体组成，在安全域中的所有对象按同种或相近的安全规则进行工作。由于各安全域之间安全策略的差异，使得它们之间的互操作的难度很大。因此，对安全域的安全策略进行形式化的描述非常重要。另外，本安全域中的各个实体如何在本域的安全管理器下进行安全互操作，以及在不同安全域之间的实体如何在安全域之间进行安全的互操作也是一个需要考虑的问题。在网络多级安全策略模型中包含安全主体、安全客体、主客体之间允许的操作以及安全信息的安全级别等。

1. 安全域的格模型

安全域由安全策略和安全实体元素组成。安全策略是为保证提供一定级别的安全性所必须遵循的规则，实体元素的一切活动必须限制在该域的安全策略范围之内。

设 E 包括系统的所有实体元素，是系统主体和客体的有限集；P 是系统的安全策略集（有限集），则安全域可用二元组表示为 $D=(E_D, P_D)$。其中 $E_D \in \text{power}(E)$ 和 $P_D \in \text{power}(P)$ 分别表示安全域 D 的安全实体集和安全策略集。全体安全域所构成的集合 SG 定义为

$$\text{SG} = \{D \mid D = (E_D, P_D), E_D \in \text{power}(E), P_D \in \text{power}(P)\}$$

$\forall D=(E_D, P_D), D'=(E_{D'}, P_{D'}) \in \text{SG}$，则元素 D 和 D' 的子域关系“$\leqslant$”可定义为

$$D \leqslant D' \Leftrightarrow (E_D \subseteq E_{D'}) \wedge (P_D \supseteq P_{D'})$$

序对 $\langle \text{SG}, \leqslant \rangle$ 是偏序集，事实上 $\forall D, D', D'' \in \text{SG}$，有：

(1) 自反性：$D \leqslant D'$

(2) 传递性：$D \leqslant D', D' \leqslant D'' \Rightarrow D \leqslant D''$

(3) 反对称性：$D \leqslant D', D' \leqslant D \Rightarrow D = D'$

考虑代数系统 $\text{SG}_m = \langle \text{SG}, \otimes, \oplus, C, (\varnothing, P), (O, \varnothing) \rangle$。其中 $(\varnothing, P)$ 和 $(O, \varnothing)$ 分别是代数系统的零元和单位元；$\forall D=(E_D, P_D), D'=(E_{D'}, P_{D'}) \in \text{SG}$，则运算 $\otimes, \oplus, C$ 的定义如下：

(1) $D \oplus D' = (E_D, P_D) \oplus (E_{D'}, P_{D'}) = (E_D \cup E_{D'}, P_D \cap P_{D'})$

(2) $D \otimes D' = (E_D, P_D) \oplus (E_{D'}, P_{D'}) = (E_D \cap E_{D'}, P_D \cup P_{D'})$

(3) $D^C = (E_D, P_D)^C = (E_D^C, P_D^C)$

不难验证代数系统 $\langle \text{SG}, \otimes, \oplus, C, (\varnothing, P), (O, \varnothing) \rangle$ 满足布尔代数的公理，且是布尔格：

$$\forall D, D' \in \text{SG}, \text{Sup}(D, D') = D \oplus D' \in \text{SG}, \text{Inf}(D, D') = D \otimes D' \in \text{SG}$$

因此，任何一个多安全域系统应该是 SG 的子集或子代数。格模型抽象地表示了系统的结构；偏序关系确定了域间的从属关系；运算涉及域间的联结关系。

2. 安全域的表示

考虑 $\forall e_i \in E, p_j \in P$，令 $P_j = P - \{p_j\}$，$D_{ip} = (\{e_i\}, P)$，$D_{oj} = (\varnothing, P_i)$，根据格的原子定义有：$D_{ip}$ 和 D_{oj} 是 SG_m 的原子。因此，由格的布尔表示定理得到：

定理 2.5.1 $\forall D \in \text{SG}, D \neq (\varnothing, P), D = \Sigma_i \Sigma_j (D_{ip} \oplus D_{oj})$，其中 $D_{ip} \leqslant D$ 和 $D_{oj} \leqslant D$，并且除顺序外，安全域 D 的表示式是唯一的。

3. 多级安全域模型

定义 2.5.1 安全级可定义为元组：$\text{SC}_m = \langle \text{SC}, \oplus, \otimes, L, H \rangle$。其中 $\text{SC} = \{\text{sc}_i \mid i = 1, 2, \cdots, n\}$ 是安全级集，$L = \text{sc}_1$ 是最低安全级；$H = \text{sc}_n$ 是最高安全级；运算 $\oplus, \otimes$ 和偏序关系“$<$”可定义为

$$\text{sc}_i \oplus \text{sc}_j = \text{sc}_{\max(i,j)}; \quad \text{sc}_i \otimes \text{sc}_j = \text{sc}_{\min(i,j)}; \quad \text{sc}_i < \text{sc}_j \Leftrightarrow i < j$$

SC 显然是一个格，并且“$<$”是一个全序关系。

定义 2.5.2 多级安全域可定义为元组：$\text{MS}_m = \langle \text{MS}, \oplus, \otimes, ((\varnothing, P), L), (O, \varnothing), H \rangle$。其中 MS 是 SG 和 SC 的直积：$\text{MS} = \text{SG} \times \text{SC} = \{(D, \text{sc}_i) \mid D \in \text{SG}, \text{sc}_i \in \text{SC}\}$。$\forall (D, \text{sc}_i), (D', \text{sc}_j) \in \text{MS}$，运算 $\oplus, \otimes$ 和偏序关系“$\leqslant$”的定义如下：

(1) $(D, sc_i) \oplus (D', sc_j) = (D \oplus D', sc_i \oplus sc_j) = ((E_D \cup E_{D'}, P_D \cap P_{D'}), sc_{\max(i,j)})$

(2) $(D, sc_i) \otimes (D', sc_j) = (D \otimes D', sc_i \otimes sc_j) = ((E_D \cap E_{D'}, P_D \cup P_{D'}), sc_{\min(i,j)})$

(3) $(D, sc_i) \leqslant (D', sc_j) \Leftrightarrow (D \leqslant D') \wedge (sc_i \leqslant sc_j)$

MS_m 显然是一个格，并且“$\leqslant$”是一个偏序关系。因此，任何一个多级安全域应该是 MS_m 的子集。该子集的元素 (D, sc) 是一个二维向量，两个分量分别表示安全域和安全级。尽管该子集不一定是格，不过由最小上界运算符 $\oplus$ 和最大下界运算符 $\otimes$ 可以构造出一个包含该多级安全域的最小格。

2.5.2 多级安全域之间的关系

1. 子域关系和对等域关系

安全域可能包含安全子域，较大的安全域将其安全策略遗传给所属的安全子域，子安全域必须能够利用其父安全域的安全策略。另外，两个通信实体要进行通信，也必须了解其所在安全域的对等关系。因此，根据安全域之间的地位和安全策略关系可以简单地将安全域之间的关系分成子域关系和对等域关系。

子域关系：这是指在两个安全域中，其中一个安全域是另外一个安全域的子域。它们体现了多级的安全策略，包括两种情况：子域的安全策略完全共享给其父域；子域的安全策略仅部分共享给其父域，子域还保留有自己独有的安全策略。

对等域关系：若两个安全域之间不是子域关系，则它们之间就是对等域关系，包括两种情况，即自治的对等域（每个域有自己的安全策略，没有相同的控制策略）和非自治的对等域（每个域有自己独有的安全策略，并且还有一个公共的安全控制策略）。

2. 相互信任域关系和非相互信任域关系

从安全域之间的信任关系来看，可以分成两种：相互信任域和不相互信任域。相互信任域之间的通信可以通过域之间的服务设施来完成，而不相互信任域之间的通信只能通过相互信任的第三方进行公正协商来完成。

2.5.3 多级安全域认证体系结构

认证和授权是分布式系统安全的基础。在 Kerberos 和 Kryptoknight 等安全认证系统中，它们重点解决的是如何实现客户端和服务器之间通过一个可信的第三方认证服务器 AS 来完成认证，它们只局限于一个特定的安全域环境，并不能满足多级安全域的认证需求。下面从安全域之间的信任关系出发对多安全域的认证体系结构进行讨论，如图 2.5.1 所示。

(1) 子安全域内的认证：图 2.5.1 中 C1 和 C2 之间的认证或 C3 和 C4 之间进行的认证。对此类子安全域内的认证，可以通过它们共同信任的本域认证代理服务器来进行密钥的分发和管理，认证可以采用 Needham-Schroeder 认证协议。

(2) 相互信任子安全域之间的认证：由于两个安全域之间是相互信任的，一般而言，信任是分级的，我们建立域内认证模型的一个前提是 C1 和 C2 信任 S1；C3 和 C4 信任 S2。如果 C1 要和另一个域 S2 的对象 C3 进行安全通信，则 C1 和 C3 之间的会话密钥由 S1 和 S2

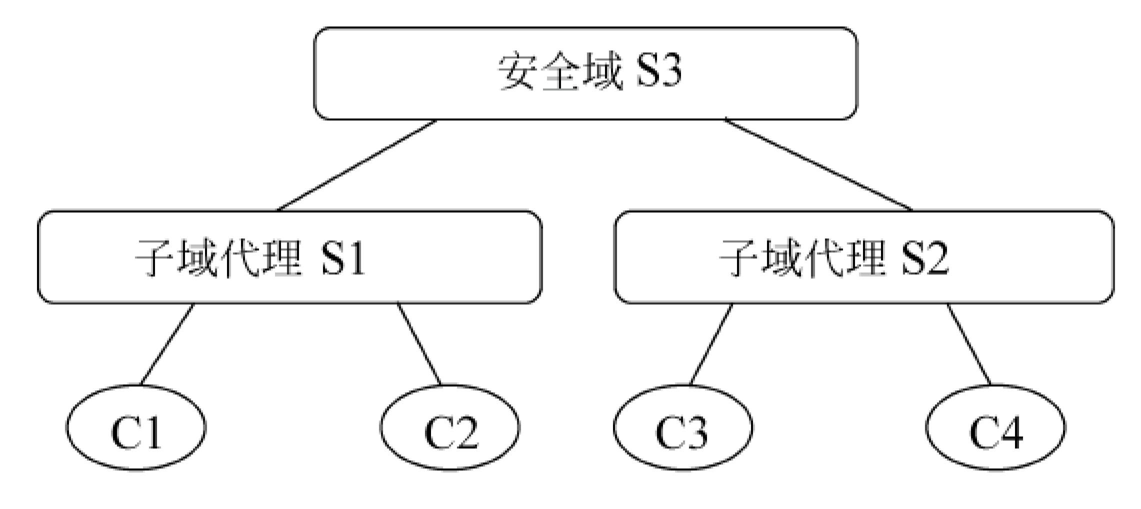


图 2.5.1　多级安全域的认证体系结构

共同协商产生。

(3) 不相互信任子安全域之间的认证：如果 S1 和 S2 不相互信任，则它们之间必须通过它们共同信任的第三方域间认证协调代理服务器来认证，即 S1 和 S2 的上级父安全域 S3。这时 C1 和 C3 之间的会话密钥是由 S1，S2 和 S3 共同协商来产生的。

2.5.4　多级安全域的认证协议

1. 信任安全域的认证协议

若两个安全域认证是相互信任的，则 R 和 S 可共同完成两个信任域中的对象之间的安全认证，如图 2.5.2 所示。

C1：$A \to R$：A, B, N_a

C2：$R \to A$：$\{N_a, K_{ab}\}_{K_{ar}}$

C3：$R \to S$：$\{A, B, N_a, K_{ab}\}_{K_{rs}}$

C4：$S \to B$：$\{A, B, N_a, K_{ab}\}_{K_{sb}}$

C5：$B \to A$：$\{N_a - 1, N_b\}_{K_{ab}}$

C6：$A \to B$：$\{N_b + 1\}_{K_{ab}}$

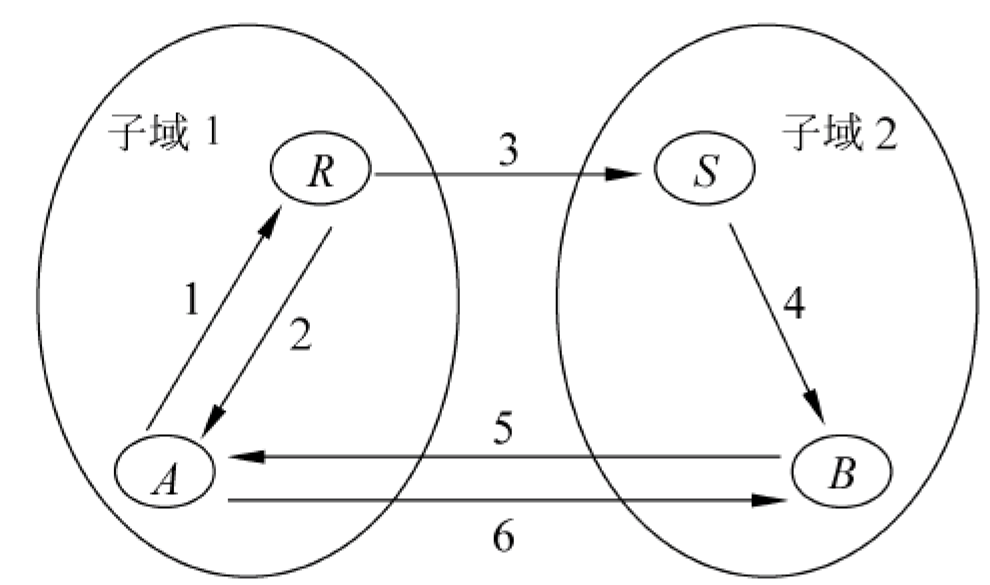


图 2.5.2　相互信任域的安全认证协议

2. 非信任安全域的认证协议

若 R 和 S 不相互信任，则必须通过 R 和 S 共同信任的第三方 T，它可以是 R 和 S 共同信任的上级父域代理认证服务器，它和 R，S 共同完成不信任域中对象之间的安全认证。如图 2.5.3 所示。

C1：$A \to R$：A, B, N_a

C2：$R \to T$：$\{N_a, A, B, R, N_r\}_{K_{rt}}$

C3：$T \to R$：$\{A, B, R, K_{rs}, N_a, N_r, \{K_{rs}, R, N_r\}_{K_{st}}\}_{K_{rt}}$

C4：$R \to S$：$\{K_{rs}, R, N_r\}_{K_{st}}$

C5：$S \to R$：$\{N_r - 1, N_s\}_{K_{rs}}$

C6：$R \to S$：$\{N_s + 1, K_{ab}\}_{K_{rs}}$

C7：$R \to A$：$\{A, B, N_a, K_{ab}\}_{K_{ar}}$

C8：$S \to B$：$\{A, B, N_a, K_{ab}\}_{K_{sb}}$

C9：$B \to A$：$\{N_a - 1, N_b\}_{K_{ab}}$

C10：$A \rightarrow B$：$\{N_b+1\}_{K_{ab}}$

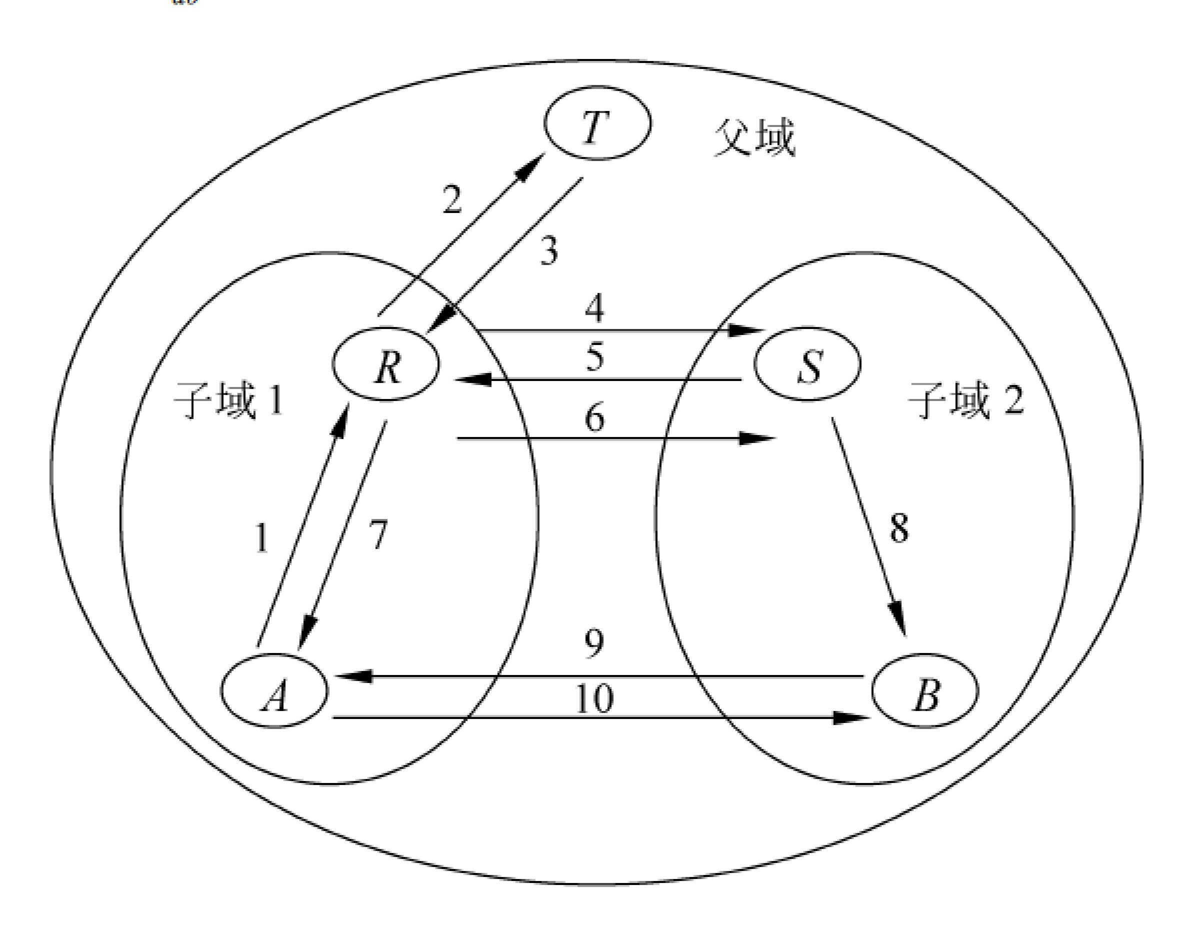


图 2.5.3 非相互信任域的安全认证协议

2.5.5 利用逻辑理论对安全域认证协议的形式化描述

2.5.5.1 BAN 逻辑

设 A,B,S 表示具体对象;K_a,K_b,K_s 表示具体的共享密钥;N_a,N_b,N_s 表示随机数;P,Q 表示任意对象;X,Y 表示任意语句,则 BAN 的逻辑语义如下。

(1) $P|\equiv X$:P 相信 X

(2) $P \triangleleft X$:P 曾经收到 X

(3) $P|\approx X$:P 曾经发送 X

(4) $P|\Rightarrow X$:P 对 X 有管辖权

(5) $\# X$:X 是新的

(6) $H(X)$:X 是单向杂凑函数

(7) (X,Y):X 和 Y 联结

(8) $P\leftrightarrow Q(K)$:P 和 Q 使用相同的密钥 K 进行报文传输

(9) $\{X\}_k$:由密钥 K 加密报文 X

BAN 逻辑主要有以下 5 条逻辑公理:

① 消息含义法则:$\dfrac{P|\equiv Q\leftrightarrow P(K),P\triangleleft X}{P|\equiv Q|\approx X}$

② 临时值校验法则:$\dfrac{P|\equiv \# X,P|\equiv Q|\approx X}{P|\equiv Q|\equiv X}$

③ 管辖法则:$\dfrac{P|\equiv Q|\Rightarrow X,P|\equiv Q|\equiv X}{P|\equiv X}$

④ 逻辑公理:$\dfrac{P\triangleleft (X,Y)}{P\triangleleft X}$;$\dfrac{P|\equiv Q\leftrightarrow P(K),P\triangleleft \{X\}_K}{P\triangleleft X}$

⑤ 逻辑公理:$\dfrac{P|\equiv \# X}{P|\equiv \#(X,Y)}$

2.5.5.2 非信任安全域认证协议的形式化证明

1. 初始假设集合

$A \mid\equiv (A \leftrightarrow R(K_{ar}))$； $R \mid\equiv (A \leftrightarrow R(K_{ar}))$； $B \mid\equiv (B \leftrightarrow S(K_{bs}))$； $S \mid\equiv (B \leftrightarrow S(K_{bs}))$；
$R \mid\equiv (R \leftrightarrow T(K_{rt}))$； $T \mid\equiv (R \leftrightarrow T(K_{rt}))$； $S \mid\equiv (S \leftrightarrow T(K_{st}))$； $T \mid\equiv (S \leftrightarrow T(K_{st}))$；
$B \mid\equiv S \mid \Rightarrow (A \leftrightarrow B(K_{ab}))$； $A \mid\equiv R \mid \Rightarrow (A \leftrightarrow B(K_{ab}))$；
$R \mid\equiv T \mid \Rightarrow (R \leftrightarrow S(K_{rs}))$； $S \mid\equiv T \mid \Rightarrow (R \leftrightarrow S(K_{rs}))$；

随机数：

$A \mid\equiv R \mid \Rightarrow \#(A \leftrightarrow B(K_{ab}))$； $A \mid\equiv \# N_a$； $A \mid\equiv \# N_b$；
$R \mid\equiv T \mid \Rightarrow \#(R \leftrightarrow S(K_{rs}))$； $R \mid\equiv \# N_r$； $S \mid\equiv \# N_s$；
$S \mid\equiv \#(R \leftrightarrow S(K_{rs}))$； $B \mid\equiv \#(B \leftrightarrow S(K_{sb}))$

2. 认证协议的预期目标集α

$A \mid\equiv (A \leftrightarrow B(K_{ab}))$； $B \mid\equiv (A \leftrightarrow B(K_{ab}))$；
$R \mid\equiv (R \leftrightarrow S(K_{rs}))$； $S \mid\equiv (R \leftrightarrow S(K_{rs}))$

也即协议的预期目标是：A 和 B 之间最终完成认证，获得子域认证服务器 R 为它们分配的会话密钥；同时，服务器 R 和 S 之间也完成认证，获得父域认证服务器 T 为它们分配的会话密钥。

3. 认证协议推理目标集合β

利用初始假设集合和逻辑公理推理。如图 2.5.3 所示，首先考虑认证协议的消息 3：

$$R \triangleleft \{N_a, N_r, R \leftrightarrow S(K_{rs}), \#(R \leftrightarrow S(K_{rs})), \{N_r, R \leftrightarrow S(K_{rs})\}_{K_{st}}\}_{K_{rt}} \quad (2.5.1)$$

利用假设 $R \mid\equiv (R \leftrightarrow T(K_{rt}))$和 BAN 逻辑公理中的①可推导出：

$$R \mid\equiv T \mid\approx N_a, N_r, R \leftrightarrow S(K_{rs}), \#(R \leftrightarrow S(K_{rs})), \{N_r, R \leftrightarrow S(K_{rs})\}_{K_{st}} \quad (2.5.2)$$

利用假设 $R \mid\equiv \# N_r$ 和逻辑公理中的②可推导出：

$$R \mid\equiv T \mid\equiv (R \leftrightarrow S(K_{rs}), \#(R \leftrightarrow S(K_{rs}))) \quad (2.5.3)$$

利用假设 $R \mid\equiv T \mid \Rightarrow (R \leftrightarrow S(K_{rs}))$和逻辑公理中的③⑤可推导出：

$$R \mid\equiv (R \leftrightarrow S(K_{rs}), \#(R \leftrightarrow S(K_{rs}))) \quad (2.5.4)$$

因此，我们可得 $R \mid\equiv (R \leftrightarrow S(K_{rs}))$和 $R \mid\equiv \#(R \leftrightarrow S(K_{rs}))$成立。

考虑认证协议的消息 4：

$$R \triangleleft \{N_r, R \leftrightarrow S(K_{rs})\}_{K_{st}} \quad (2.5.5)$$

利用假设 $S \mid\equiv (S \leftrightarrow T(K_{st}))$和 $S \mid\equiv \#(R \leftrightarrow S(K_{rs}))$以及逻辑公理中的①②可推导出：

$$S \mid\equiv T \mid\equiv (R \leftrightarrow S(K_{rs})) \quad (2.5.6)$$

利用假设 $R \mid\equiv T \mid \Rightarrow (R \leftrightarrow S(K_{rs}))$和逻辑公理中的③可推导出：

$$S \mid\equiv (R \leftrightarrow S(K_{rs})) \quad (2.5.7)$$

根据式(2.5.4)和式(2.5.7)可知，服务器 R 和 S 之间完成认证，并获得认证服务器 T 为它们分配的会话密钥 K_{rs}。

考虑认证协议的消息 7：

$$A \triangleleft \{N_a, A \leftrightarrow B(K_{ab}), \#(A \leftrightarrow B(K_{ab}))_{K_{ar}}\} \quad (2.5.8)$$

利用假设 $A \mid\equiv (A \leftrightarrow R(K_{ar}))$和 $A \mid\equiv \# N_a$ 以及逻辑公理中的①②可推导出：

$$A\mid\equiv R\mid\equiv(A\leftrightarrow B(K_{ab}),\#(A\leftrightarrow B(K_{ab})))\tag{2.5.9}$$

利用假设 $A\mid\equiv R\mid\Rightarrow\#(A\leftrightarrow B(K_{ab}))$和逻辑公理中的③⑤可推导出：

$$A\mid\equiv(A\leftrightarrow B(K_{ab}),\#(A\leftrightarrow B(K_{ab})))\tag{2.5.10}$$

因此，我们可得 $A\mid\equiv(A\leftrightarrow B(K_{ab}))$和 $A\mid\equiv\#(A\leftrightarrow B(K_{ab}))$成立。

考虑认证协议的消息 8：

$$B\lhd\{N_a,A\leftrightarrow B(K_{ab})\}_{K_{sb}}\tag{2.5.11}$$

利用假设 $B\mid\equiv(B\leftrightarrow S(K_{bs}))$和 $B\mid\equiv\#(B\leftrightarrow S(K_{bs}))$以及逻辑公理中的①②可推导出：

$$B\mid\equiv S\mid\equiv(A\leftrightarrow B(K_{ab}))\tag{2.5.12}$$

利用假设 $B\mid\equiv S\mid\Rightarrow\#(A\leftrightarrow B(K_{ab}))$(……)和逻辑公理中的③可推导出：

$$B\mid\equiv(A\leftrightarrow B(K_{ab}))\tag{2.5.13}$$

根据式(2.5.10)和式(2.5.13)可知，A 和 B 之间完成认证，并获得认证服务器 R 为它们分配的会话密钥 K_{ab}。

4. 结论

推理目标集合 β 为

$$A\mid\equiv(A\leftrightarrow B(K_{ab}));\quad B\mid\equiv(A\leftrightarrow B(K_{ab}));$$
$$R\mid\equiv(R\leftrightarrow S(K_{rs}));\quad S\mid\equiv(R\leftrightarrow S(K_{rs}))$$

可见有预期目标集合 $\alpha\subseteq\beta$，协议达到预期目的。

参 考 文 献

1 Network Working Group. RFC2138 Remote Authentication Dial in User Service (RADIUS). April 1997

2 Network Working Group. RFC2139 RADIUS Accounting. April 1997

3 Network Working Group. RFC2865 Remote Authentication Dial in User Service (RADIUS). June 2000

4 Network Working Group. RFC2866 RADIUS Accounting. June 2000

5 Network Working Group. RFC2867 RADIUS Accounting Modifications for Tunnel Protocol Support. June 2000

6 Network Working Group. RFC2868 RADIUS Attributes for Tunnel Protocol Support. June 2000

7 Network Working Group. RFC3575 IANA Considerations for RADIUS. July 2003

8 Network Working Group. RFC1334 PPP Authentication Protocols. October 1992

9 Network Working Group. RFC2869 RADIUS Extensions. June 2000

10 Network Working Group. RFC3748 Extensible Authentication Protocol (EAP). June 2004

11 Comer D E, Stevens D L. Internetworking with TCP/IP. Vol Ⅲ. Client-Server Programming And Application: Linux/POSIX sockets version (赵刚，林瑶，蒋慧等译. 用 TCP/IP 进行网际互连 第三卷，客户-服务器编程与应用：Linux/POSIX 套接字版. 北京：电子工业出版社，2001)

12 AAA Working Group. RFC3588 Diameter Base Protocol. September 2003

13 Network Working Group. RFC4005 Diameter Network Access Server Application. August 2005

14 Network Working Group. RFC4072 Diameter Extensible Authentication Protocol (EAP) Application. August 2005

15 Network Working Group. RFC4004 Diameter Mobile IPv4 Application. August 2005

16 AAA Working Group. Internet-Draft Diameter CMS Security Application. March 2002

17　Network Working Group. RFC4006 Diameter Credit-Control Application. August 2005
18　Network Working Group. RFC2977 Mobile IP Authentication, Authorization, and Accounting Requirements. October 2000
19　Koien G M, Haslestad T. Security aspects of 3G-WLAN interworking. IEEE Communications Magazine, 2003, 41(11): 82～88
20　ETSI TR 101 957 V1.1.1. Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Equirements and Architectures for Interworking between HIPERLAN/2 and 3rd Generation Cellular systems. 2001
21　Buddhikot M, Chandranmenon G, Han S, Lee Y W, Miller S, Salgarelli L. Integration of 802.11 and third-generation wireless data networks. In: Proc of the IEEE INFOCOM 2003, Vol. 1, 2003. 503～512
22　Mahapatra A, Uma R. Authentication in an intergrated 802.1x-based WLAN and CDMA 2000-1x network. In: Proc of the IEEE APCC 2003, Vol. 1, 2003, 227～231
23　3GPP TS 23.234 V6.1.0. 3GPP System to Wireless Local Area Network (WLAN) Interworking; System Description. 2004
24　3GPP TR 22.934 V6.2.0. Feasibility Study on 3GPP System to Wireless Local Area Network (WLAN) Interworking. 2003
25　Kim H, Afifi H. Improving mobile authentication with new AAA protocols. In: Proc of the IEEE ICC 2003, 2003, Vol. 1, 497～501
26　Salkintzis A K. Interworking techniques and architectures for WLAN/3G integration toward 4G mobile data networks. IEEE Wireless Communications, 2004, 11(3): 50～61
27　Kambourakis G, Rouskas A, Kormentzas G, Gritzalis S. Advanced SSL/TLS-based authentication for secure WLAN-3G interworking. IEE Communications Proceedings, 2004, 151: 501～506
28　Aboba B, Simon D. PPP EAP TLS Authentication Protocol. IETF RFC 2716, October 1999
29　Funk P, Blake-Wilson S. EAP Tunneled TLS Authentication Protocol. IETF draft-funk-eap-ttls-v0-00.txt, February 2005

Chapter

第 3 章

数字签名

针对当前计算机网络的安全隐患和屡受攻击的现状，计算机界已经发展了许多与网络安全相关的技术。例如，使用数据加密、存取控制等许多技术可以对数据通信时的保密性和完整性予以保证。然而仅仅有这些还是不够的，特别是近年来，随着电子商务的发展，人们通过通信网络进行快速、远距离的贸易，数字或电子签名也应运而生并开始用于商业通信系统，诸如在电子邮件、电子转账、办公自动化等系统中。这些都要求根据不同的情况设计出适合特定情况的、安全而有效的数字签名，以适应飞速发展的网络环境下的安全需要。

本章对公钥密码体制和典型数字签名机制进行简单的介绍，然后详细说明基于椭圆曲线的密码体制，并提出基于椭圆曲线密码体制的群体导向(t, n)门限签名方案，给出了其安全和性能分析。

3.1 公钥密码体制

公钥密码体制由 Diffie 和 Hellman[1,2]于 1976 年在他们发表的《密码学的新方向》一文中首次提出，引发了密码学领域的一场变革。他们指明了实现在某些已知的数学求解问题上建立密码的具体途径，这使得在网络传送过程中的发送端和接收端的无密钥传输的保密通信是可能的。之后，Rivest，Shamir 和 Adleman[3]三位学者于 1977 年提出了第一个比较完善的公钥密码体制，这就是著名的 RSA 公钥密码体制。该算法允许不曾联系的两个个体之间进行保密通信。RSA 既可以用于保密，也可以用于数字签名。随后，人们又提出了各种基于不同的计算问题的公钥密码体制，如著名的 EIGamal[4]公钥密码体制。

3.1.1 密码体制分类

根据密钥的特点，Simmons[5]把密码体制分为对称密码体制（symmetric cryptosystem）和非对称密码体制（asymmetric cryptosystem）两种。

1. 对称密码体制

所谓对称密码体制，即加密和解密使用相同的密钥，因此该体制又称为单密钥密码体

制。在这种体制中,即使有时加密、解密密钥不相同,但它们之间仍存在着一定的联系,是容易互相推导出来的,如 DES 密码体制。

对称密码体制存在着两个主要的问题:一是在密钥的分配与管理方面,由于其加密与解密使用的是相同的密钥,发送和接收双方都必须知道同一个密钥,因此双方必须事先通过某一秘密途径把密钥传送到另一方。按照这种情况,当同时有 n 个用户要进行通信时,需要的密钥数为 $n(n-1)/2$ 个。因此,有关密钥的分配、安全传送、保密管理是一件很困难的事情。

此外,在数据的完整性保护方面,由于保密通信双方都有相同的加/解密密钥,信息的接收方可以很容易地篡改原文内容,信息的发送方也可以否认发出的内容。因此,对称密码体制在数字签名和身份认证上的应用是难以实现的。

2. 非对称密码体制

所谓非对称密码体制,即加密和解密使用不同的密钥,也称其为双密钥密码体制。非对称密码体制的基本思想是:不仅公开加密算法,而且加密用的密钥也公开。即可以将每一个用户的加密密钥作为公钥,而把解密密钥(私钥)保密就可以了,而且各用户的个人解密密钥由各自保密保管。若用户 A 要向用户 B 发送信息,A 只要查找到 B 已公开的公钥,并对文件进行加密后发送给用户 B 即可,而 B 在收到密文后利用个人私钥对密文进行解密即可得到由 A 发过来的明文。由于这种密码体制的加密密钥是公开的,因此又称这种加密体制为公开密钥密码体制,简称公钥体制,目前被广泛应用在数字签名与身份认证领域。

3.1.2 公钥密码体制原理

公钥密码体制的最大特点是采用两个相关密钥将加密和解密功能分开,其中一个密钥称为公钥,另一个密钥称为私钥。而且从一个密钥难以推导出另一个,且可以分离使用。通信双方在通信之前无需事先交换密钥就可以建立起保密通信,并进行数据传送。如图 3.1.1 所示。

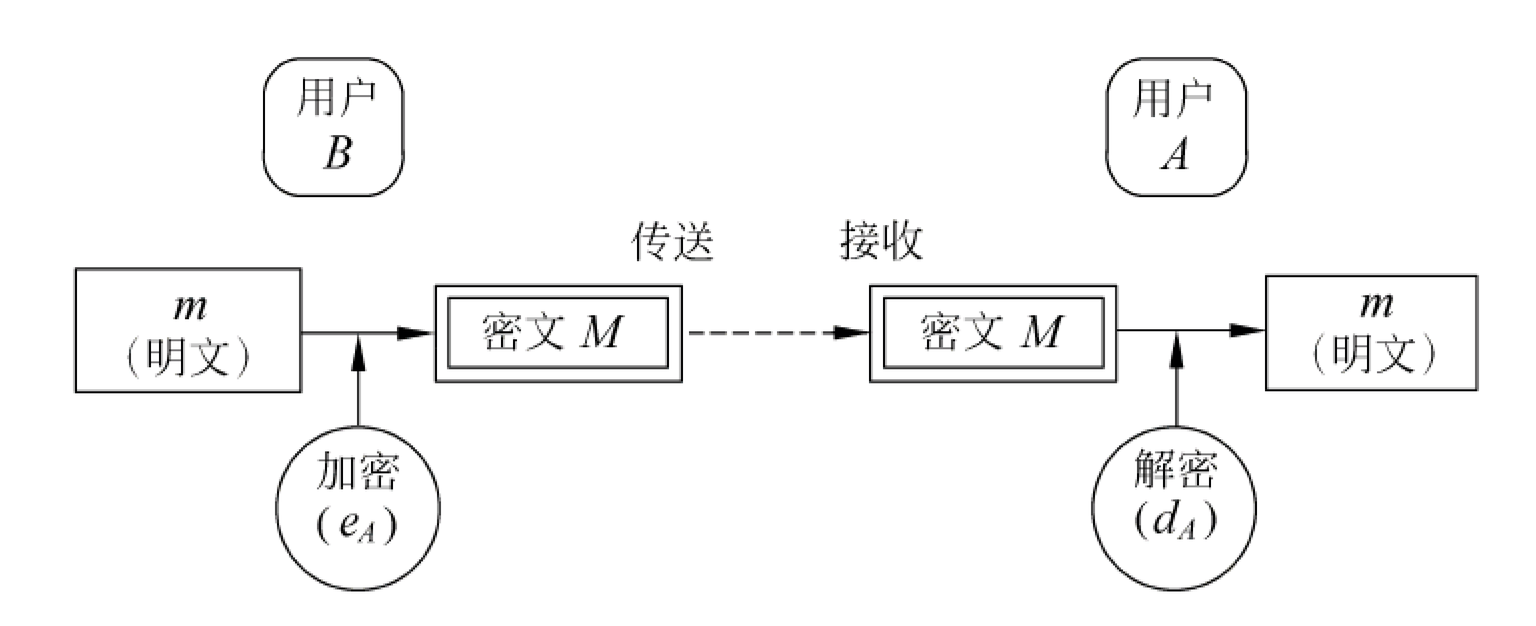


图 3.1.1 公钥密码体制

图 3.1.1 中,假设 B 要向 A 传送一个文件,其具体实现原理说明如下:

(1) 用户 A 用某种数学算法产生一对个人私钥(d_A)和公钥(e_A)。并把公钥 e_A 公布出去。

(2) 用户 B 用 A 公开的公钥 e_A 对明文 m 进行加密:$M=Ee_A(m)$,并把密文 M 发送给

用户 A。

(3) 用户 A 在收到密文 M 后,用个人私钥 d_A 恢复出明文 $Dd_A(M)=Dd_A(Ee_A(m))=m$。

由上述原理可以看出,这里的公钥 e_A 并不需要进行保密,只要保证它的真实性即可。

公钥密码体制主要可以提供以下功能。

(1) 机密性:保证非授权人员不能非法获取信息,要通过数据加密来实现。

(2) 认证:保证对方属于所声称的实体,主要通过数字签名来实现。

(3) 数据完整性:保证信息内容不被篡改,使入侵者不可能用假消息代替合法消息,主要通过数字签名来实现。

(4) 不可抵赖性:发送者不可能事后否认他发送过的消息,消息的接收者可以向中立的第三方证实所指的发送者确实发出了消息,通过数字签名来实现。

公钥密码体制的安全性基于复杂的数学难题[6]。对于某种数学难题,如果利用通用的算法计算出密钥的时间越长,那么基于这一数学难题的公钥密码体制就被认为是越安全的。目前,根据所基于的数学难题来分类,以下 3 种密码体制被认为是安全和有效的:

(1) 基于整数因子分解的密码体制,如 RSA 和 DSA 密码体制。

(2) 基于离散对数问题的密码体制,如 ElGamal[4] 密码体制。

(3) 基于椭圆曲线离散对数问题的密码体制。

公钥密码体制采用的加密密钥(私钥)、解密密钥(公钥)是不同的。由于加密密钥是公开的,密钥的分配和管理就很简单,而且能够很容易地实现数字签名,因此非常适合应用于电子商务。从本质上看,由于公钥密码体制基于尖端的数学难题,计算起来非常复杂,所以,公钥密码比私钥(如 DES 和 RC5 等)加密的速度要慢。在实际应用中,公钥密码体制并没有完全取代私钥密码体制,通常是利用二者各自的优点,采用公钥密码体制作密钥加密(或只加密少量数据),私钥密码体制则用作对大量数据进行加密,这就是混合加密体制。混合加密体制较好地解决了运算速度问题和密钥分配管理问题。

3.1.3 Diffie-Hellman 密钥交换

Diffie 和 Hellman 公钥密码体制[1,2]主要描述了通信双方的密钥交换协议。通过这一协议,通信双方 A 和 B 可以从一个不安全的传输信道上获得和分享一些秘密信息,可以将这些信息作为私钥体制中的密钥,如图 3.1.2 所示。

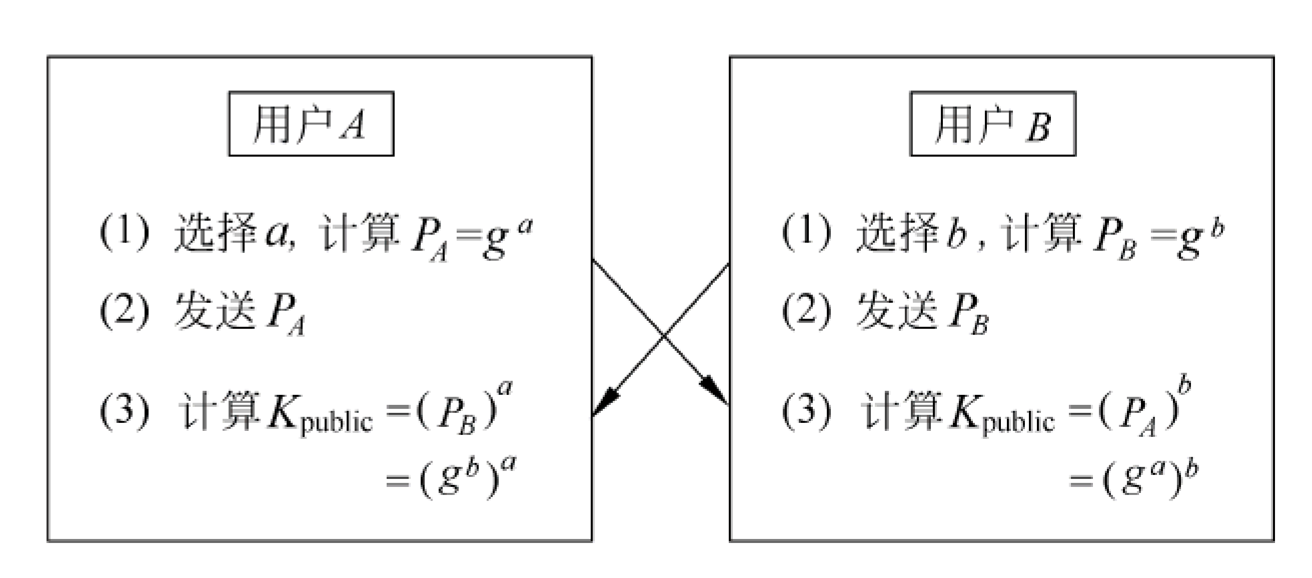


图 3.1.2 Diffie-Hellman 密钥交换

Diffie-Hellman 密钥交换过程如下：

(1) A 和 B 公开选定一个确定的群 G 和其中的一个元素 $g \in G$；

(2) A 产生一个随机整数 $a(a \in G)$，计算 $P_A = g^a$，并通过公共信道传递给 B；

(3) B 产生一个随机整数 $b(b \in G)$，计算 $P_B = g^b$，并将 g^b 传送给 A；

(4) A 收到 P_B 后，计算 $(P_B)^a = (g^b)^a$；

(5) B 收到 P_A 后，计算 $(P_A)^b = (g^a)^b$。

这样，A 和 B 就共同构建了一个共同的群元素 $P_{\text{public}} = (P_A)^b = (P_B)^a = g^{ab}$。

值得注意的是，这并不是原始的密钥交换，因为第三方 C 可以模仿 A 或 B。但是，这一协议可以用下面的方法得到修正：依靠一个重要的、值得信赖的权威来确认 g^a 来自 A 以及 g^b 来自 B，确认可以利用数字签名技术来实现。注意到攻击者 C 可以知道群 G，g，g^a 和 g^b，并利用这些信息来构建出 g^{ab}。这一问题一般称为 Diffie-Hellman 问题。

由上文可知，Diffie-Hellman 问题是建立在求解离散对数问题之上的。如果攻击者 C 在得知 g 和 g^a 的知识后，通过求解离散对数问题从而求出整数 a，那么攻击者 C 就成功地破解了 Diffie-Hellman 问题。

3.1.4　RSA 密码体制

RSA 密码体制用数论构造，其理论基础是一种特殊的可逆模指数运算。它的安全性基于分解大整数困难性，是迄今为止理论上最为成熟的公钥密码体制，该体制目前被广泛应用。下面是 RSA 算法[5]的描述：

设 n 是两个大素数 p 和 q 的积，即 $n = pq$，$\phi(n) = (p-1)(q-1)$，$K = \{(n, p, q, e, d) \mid n = pq, ed \equiv 1(\bmod\ \phi(n))\}$。对每一个 $K = (n, p, q, e, d)$ 定义如下。

加密过程：$E_K(x) = x^e(\bmod\ n)$，$x \in Z_n$；

解密过程：$D_K(y) = y^d(\bmod\ n)$，$x \in Z_n$，

其中 n 和 e 被公开，而 p 和 d 保密。

为了建立密码系统，签名方用户 B 需要完成以下步骤：

(1) 随机选取两个大素数 p 和 q；

(2) 计算 $n = pq$ 和 $\phi(n) = (p-1)(q-1)$；

(3) 随机选取整数 e，使其满足 $\gcd(e, \phi(n))$，且 $1 < e < \phi(n)$；

(4) 计算 d，使其满足 $ed \equiv 1(\bmod\ \phi(n))$，并作为解密密钥。

最后将 n 和加密密钥 e 公开。

由上述算法可知，RSA 密码体制的安全性建立在两个大素数 p 和 q 上，假若 $n = pq$ 能够成功地被因式分解，则 RSA 便被击破。因为若 p 和 q 为已知，则 $\phi(n) = (p-1)(q-1)$ 便可被计算出，而解密密钥 d 满足 $ed \equiv 1(\bmod\ \phi(n))$，故 d 很容易求得。因此，RSA 的安全性依赖于因式分解的困难性。随着计算水平的不断提高，为了加强 RSA 算法的安全性，只好不断增加 RSA 算法的密钥长度。在电子商务的 SET 协议中，规定用户使用 1024 比特或以上的 RSA 密钥，而认证中心 CA 则需要使用 2048 比特或以上的 RSA 密钥。但这样做会导致运算速度（特别是解密时）的下降，以及密钥存储和管理成本的增加。

3.1.5 ElGamal 密码体制

与 RSA 一样，ElGamal 算法在密码协议中也是被广泛应用的一类公钥密码算法，而最初的椭圆曲线密码体制也是基于 ElGamal 算法原理的。ElGamal 算法的安全性是基于求解离散对数问题(DLP)的困难性。以下是 ElGamal 算法的详细描述。

1. 参数的构成

首先使用者 A 和 B 按如下方法各自生成一对个人公钥和私钥：

(1) 生成一个大的随机素数 p 和整数模 p 的乘法群 Z_n^* 的一个生成元 α；

(2) 选取一个随机整数 a，$1 \leqslant a \leqslant p-2$，计算 $\alpha^a \pmod p$；

(3) 用户 A 的公开密钥是 (p, α, α^a)；私钥是 a。

2. 加密过程

现在假设用户 B 要对消息 m 加密并发送给 A，则用户 B 执行如下步骤：

(1) 获取 A 的公开密钥 (p, α, α^a)；

(2) 将消息 m 表示成 $(0, 1, \cdots, p-1)$ 范围内的整数；

(3) 选取一个随机整数 k，$1 \leqslant k \leqslant p-2$；

(4) 计算 $\gamma = \alpha^k \pmod p$ 和 $\delta = (m(\alpha^a)^k) \pmod p$；

(5) 发送密文 $M = (\gamma, \delta)$ 给 A。

3. 解密过程

A 在收到密文 M 后，执行如下步骤进行解密：

(1) 用个人私钥 e 计算出 $\gamma^{p-1-a} \pmod p = \gamma^{-a} = \alpha^{-ak}$；

(2) 计算 $\gamma^{-a} \cdot \delta \pmod p \equiv \alpha^{-ak} m \alpha^{-ak} \equiv m \pmod p$ 恢复出明文 m。

由上述加解密过程可知，ElGamal 密码体制也是建立在解离散对数问题(DLP)困难度之上的。有一点需要指出的是，在 ElGamal 密码体制中要用不同的随机整数 k 来加密不同的消息。因为假若使用同一个随机整数 k 来加密两个消息 m_1 和 m_2，所得到的密文对分别是 (γ_1, δ_1) 和 (γ_2, δ_2)，$\delta_1/\delta_2 = m_1/m_2$，故当 m_1 为已知时，m_2 也可以计算出来。另外，ElGamal 密码体制还有一个称为"消息扩展"的缺点，即密文长度是所对应的明文长度的两倍。

3.2 数字签名

数字签名是建立在公钥密码体制上的一种应用，它在网络安全，包括身份认证、数据完整性、不可否认以及匿名方面有着重要的应用。本节主要介绍数字签名的基本概念和几个特殊的数字签名方案。

3.2.1 数字签名基本概念

为了鉴别文件或书信的真伪性，传统的做法是相关人员在文件或书信上亲笔签名或印章。而数字签名并不是使用"手写签名"类型的图形标志，它利用各种加密算法来完成签名

的功能。数字签名用来保证信息传输过程中信息的完整性并提供信息发送者的身份。在电子商务中安全、方便地实现在线支付，而数据传输的安全性、完整性、身份验证机制以及交易的不可抵赖措施等大多通过安全性认证手段加以解决，电子签名可以进一步方便企业和消费者在网上做生意，使企业和消费者双方获利。例如，商业用户无需在纸上签字或为信函往来而等待，足不出户就能通过网络获得抵押贷款、购买保险或者与房屋建筑商签订契约等；企业之间也能通过网上协商达成有法律效力的协议。

基于公钥密码体制和私钥密码体制都可以获得数字签名，但目前几乎所有的数字签名都是基于公钥密码体制的。数字签名原理如图 3.2.1 所示，其处理过程如下：

(1) 使用单向散列函数将发送文件加密产生数字摘要；

(2) 发送方用自己的私钥对摘要再加密，这样就形成了数字签名；

(3) 将原文和加密的摘要同时传给对方；

(4) 接收方用发送方的公共密钥对摘要解密，同时对收到的文件用同样的单向散列函数加密产生又一个摘要；

(5) 将解密后的摘要和收到的文件在接受方与重新产生的摘要对比，如果两者一致，则说明是发送方签名的文件，而且传送过程中信息没有被破坏和篡改，否则，或者不是发送方签的名，或者信息已失去其安全性和保密性。

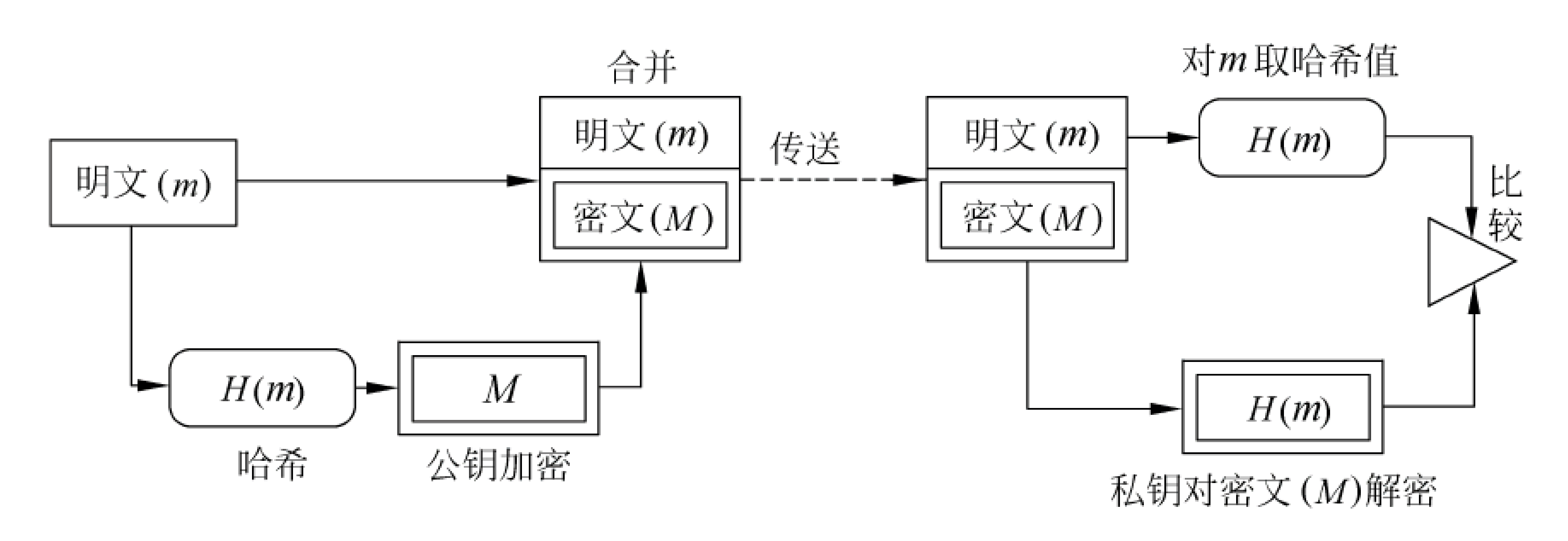


图 3.2.1　数字签名原理

3.2.2　数字签名的特点

随着信息时代的来临，人们希望通过数字通信网络进行迅速的、远距离的贸易合同的签名，数字或电子签名法应运而生，并开始用于商业通信系统，诸如电子邮递、电子转账、办公自动化等系统中。

手写签名与数字签名的主要区别在于：

(1) 签署文件方面的不同。一个手写签名是所签文件的物理部分，而一个数字签名并不是所签文件的物理部分，因此所使用的数字签名算法必须设法把签名“绑”到所签文件上。

(2) 验证方面的不同。一个手写签名是通过与一个真实的手写签名相比较来验证的，这种方法很不安全，且很容易伪造。而数字签名能够通过一个公开的验证算法来验证，这样，任何人都能验证数字签名，安全的数字签名算法的使用将阻止伪造签名的可能性。

(3) 复制方面的不同。一个手写签名不易复制，因为一个文件的手写签名的复制容易与原文件区别开来。而一个数字签名容易复制，因为一个文件的数字签名的复制与原文件一样，这个特点要求我们阻止一个数字签名的重复使用，一般通过要求信息本身包含诸如日

期等信息来达到阻止重复使用签名的目的。

一个签名算法至少应满足以下 3 个条件：

(1) 签名者事后不能否认自己的签名；

(2) 接收者能够验证签名，而其他任何人都不能伪造签名；

(3) 当双方关于签名的真伪性发生争执时，一个法官或第三方能够解决双方之间发生的争执。

从接收者验证签名的方式可将数字签名分为真数字签名和仲裁数字签名两大类。在真数字签名中，签名者直接把签名消息发送给接收者，接收者无需求助第三方即能验证签名。而在仲裁签名中，签名者把签名消息经被称为仲裁者的第三方发送给接收者，接收者不能直接地验证签名，签名的合法性是通过仲裁者作为媒介来保证的，也就是说，接收者要验证签名必须与仲裁者合作。

从计算能力上看，可将数字签名分为无条件安全的数字签名和计算上安全的数字签名。现有的数字签名大部分都是计算上安全的，诸如 RSA 数字签名、ElGamal 数字签名等。所谓计算上安全的数字签名是指任何有足够能力的伪造者总能伪造签名者的签名。无条件安全的数字签名的签名者和接收者都是无条件安全的。理论上，它们在许多应用中能够代替计算机上安全的数字签名，但在实际应用中是不太有效且不能被应用的，这是因为，在这种数字签名中需要一个复杂的交互密钥生成协议，而且签名很长，如同公钥密码体制的情况，我们的主要目的还是要设计计算上安全的数字签名方案。

3.2.3 RSA 数字签名体制

RSA 数字签名是以 RSA 加密算法来实现的。总的来说，就是签名方使用个人私钥 d 对消息进行加密（签名）；验证方使用签名方公开的公钥 e 解密并认证。下面将对 RSA 数字签名和认证过程进行描述。

1. 签名过程

(1) 使用 Hash 函数将消息产生数字摘要 $H(m)$；

(2) 使用个人私钥 d 对哈希后的消息 $H(m)$ 进行加密（签名）：$M=(H(m))^d \bmod n$；

(3) 将明文消息 m 和加密密文 M 同时发送给验证方。

需要指出的是，当明文消息 m 很短时，可直接对消息加密 $M=m^d \bmod n$ 而无需再取哈希。

2. 验证过程

当验证方接收到 (M,m) 后，按照如下步骤来验证签名的有效性：

(1) 获得签名方的公钥 e；

(2) 对密文进行解密：$h=M^e \bmod n$；

(3) 对收到的明文消息 m 进行哈希取值 $H(m)$；

(4) 判别 h 与 $H(m)$ 是否相等。如果等式 $h=H(m)$ 成立，则代表签名有效；否则，签名无效。

在上述签名方案中，首先对消息 m 进行哈希(hash)后再加密，这样可以有效地保护公钥 e，同时有身份辨别和保持数据完整性的功能，达到数字签名的要求。

3. RSA 算法证明

在该算法的证明过程中，主要用到了模运算的一些性质。以下是证明过程：

已知：$M=(H(m))^d \bmod n$；证明：$H(m)=h=M^e \bmod n$。

证明：$h=M^e \bmod n$

$$\Leftrightarrow h = ((H(m))^d \bmod n)^e \bmod n$$
$$\Leftrightarrow h = (H(m))^{de} \bmod n$$
$$\Leftrightarrow h = (H(m))^{k\phi(n)+1} \bmod n \quad (\text{因为 } ed = 1 \bmod \phi(n))$$
$$\Leftrightarrow h = H(m) \bmod n \qquad \square$$

3.2.4　ElGamal 数字签名体制

ElGamal 算法既可以用于数字签名，也可以用于加密和解密，其安全性建立在计算有限离散对数问题（DLP）的困难度上。目前已被 ANSI X9.30-199X 采纳为数字签名的标准算法。

假定待签名消息为 m。下面将对 ElGamal 数字签名体制进行描述：

1. 参数初始化

(1) 在 Z_p 选取一个大素数 p；

(2) 在乘法群 Z_p^* 中选定一个生成元 g；

(3) 选取一个 x，$x\in Z_p^*$ 作为私钥；

(4) 计算公钥 $y=g^x \bmod p$，并把 y 公布。

2. 签名过程

(1) 随机选取一个 k，$k\in Z_p^*$；

(2) 计算 $r=g^k \bmod p$；

(3) 计算 $s=(H(m)-xr)k^{-1} \bmod (p-1)$；

(4) 把消息 m 和签名$\{r,s\}$发送给验证方。

3. 验证过程

验证方接收到消息 m 和签名$\{r,s\}$后，按以下步骤验证签名的有效性：

(1) 验证 r 是否满足 $1\leqslant r\leqslant p-1$。若成立，则继续下一步；否则，签名是不合法的；

(2) 计算 v_1：$v_1=y^r r^s \bmod p$；

(3) 计算 v_2：$v_2=g^{H(m)} \bmod p$；

(4) 验证 $v_1=v_2$ 是否成立。若成立，则签名有效；否则，签名无效。

值得注意的是，在上述签名过程中引入了随机变量 k，这使得对于同一个待签名，由于 k 的不同而导致签名$\{r,s\}$的不同。另外，可以看到待消息的空间为 Z_p^*，而签名结果的空间则为 $Z_p^*\times Z_{p-1}$。

4. ElGamal 算法证明

由于 $s=(H(m)-xr)k^{-1} \bmod (p-1)$

$\Leftrightarrow k\cdot s=(H(m)-xr) \bmod (p-1)$

$\Leftrightarrow H(m)=ks+xr \bmod (p-1)$

所以，等式 $v_2=g^{H(m)} \bmod p$

$\Leftrightarrow v_2=g^{ks+xr \bmod (p-1)} \bmod p$

$\Leftrightarrow v_2=(g^k)^s \cdot (g^x)^r \bmod p$

$\Leftrightarrow v_2=r^s \cdot y^r \bmod p$ （因为 $y=g^x \bmod p, r=g^k \bmod p$）

$\Leftrightarrow v_2=v_1$ □

3.2.5 Schnorr 数字签名体制

Schnorr 数字签名体制[7,8]是 ElGamal 签名体制的变形，由 Schnorr 于 1989 年提出。其安全性是建立在求解离散对数困难度的基础上的。

假定待签名消息为 m，其签名验证过程如下：

1. 参数初始化

（1）在 Z_p 选取一个大素数 p 和 q。其中 $p|(q-1)$；p 大于 152 位的整数；q 大于 150 位的整数；

（2）在乘法群 Z_p^* 中选定一个生成元 g，且 $g^q=1 \bmod p$；

（3）选取一个 x，$1<x<q$ 作为私钥；

（4）计算公钥 $y=g^x \bmod p$；

（5）公布参数 p,q,g 和公钥 y。

2. 签名过程

（1）随机选取一个 k，$k\in Z_p^*$；

（2）计算 $r=g^k \bmod p$；

（3）计算 $e=H(r \| m)$；

（4）计算 $s=k+xe \bmod q$；

（5）把消息 m 和签名 $\{s,e\}$ 发送给验证方。

3. 验证过程

验证方接收到消息 m 和签名 $\{s,e\}$ 后，按以下步骤对签名进行验证：

（1）计算 r'：$r'=g^s y^{-e} \bmod p$；

（2）计算 e'：$e'=H(r' \| m)$；

（3）验证 $e=e'$ 是否成立。如果成立，则签名有效；否则，签名无效。

由上述签名验证过程可以看出，其储存空间为 Z_p^*，而签名结果的空间为 $Z_p^* \times Z_p$。

4. Schnorr 算法证明

由于 $r'=g^s y^{-e} \bmod p$

$\Leftrightarrow r'=g^s (g^x)^{-e} \bmod p$

$\Leftrightarrow r'=g^{s-xe} \bmod p$

$\Leftrightarrow r'=g^k \bmod p$

$\Leftrightarrow r'=r$

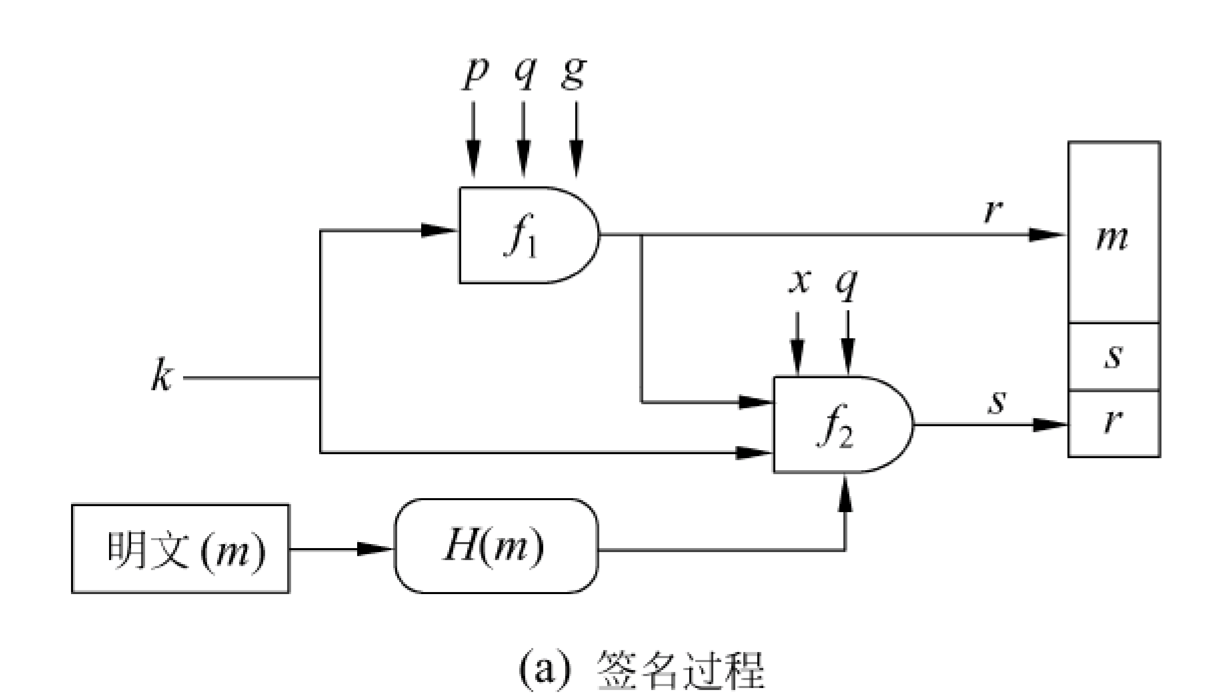
所以 $e'=H(r'\|m)$

$\Leftrightarrow e'=H(r\|m)$

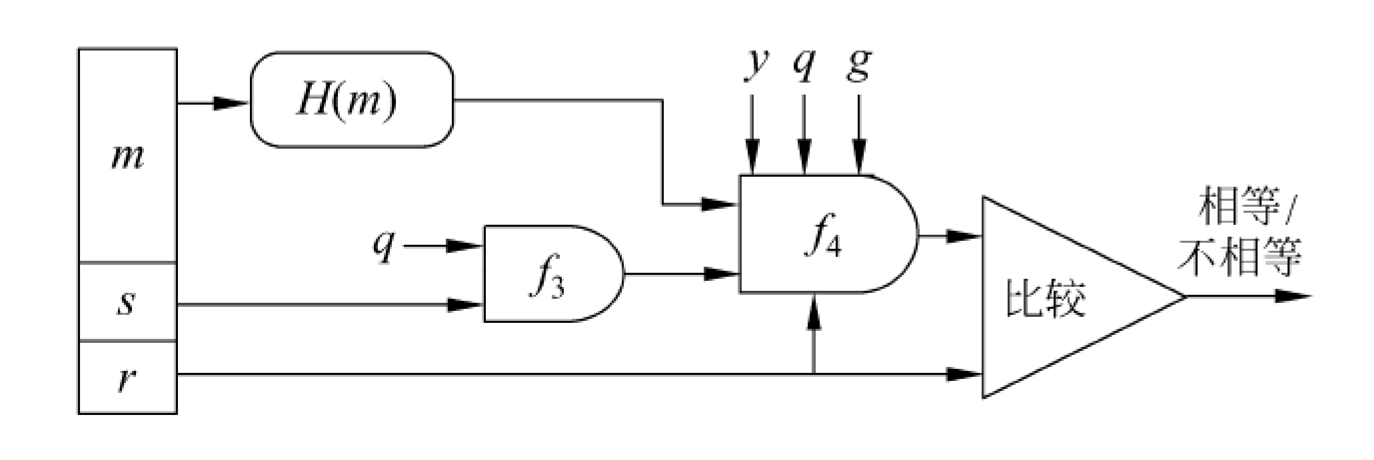
$\Leftrightarrow e'=e$ □

3.2.6　DSS 数字签名体制

DSS(digital signature standard)数字签名体制是由美国国家标准技术研究所(NIST)于 1991 年颁布,该体制采用了 DSA(digital signature algorithm)算法。该算法是 ElGamal 和 Schnorr 签名体制的变形,由 D. W. Kravitz 所设计,其安全性也是建立在求解离散对数困难度的基础上。图 3.2.2 为 DSA 签名和验证过程。



(a) 签名过程



(b) 验证过程

图 3.2.2　DSA 框图

假定待签名消息为 m。使用 DSA 算法的签名验证过程如下:

1. 参数初始化

(1) 选取一个大素数 p,$2^{L-1}<p<2^{L}$;其中,$512\leqslant L\leqslant 1024$;

(2) 选定一个$(p-1)$的素因子 q,$2^{159}<p<2^{160}$,即字长为 160 位;

(3) 计算 $g=h^{(p-1)/q}\bmod p$,其中 h 是一个整数,$1<h<(p-1)$,且 $h^{(p-1)/q}\bmod p>1$;

(4) 选取一个随机数 x,$0<x<q$ 作为私钥;

(5) 计算公钥 $y=g^{x}\bmod p$;

(6) 公布参数 p,q,g 和公钥 y。

2. 签名过程

(1) 随机选取一个 k,$0<k<q$;

(2) 计算 $r=(g^{k}\bmod p)\bmod q$,亦即 f_1;

(3) 计算 $s=\{k^{-1}[H(m)+xr]\}\bmod q$，亦即 f_2；

(4) 把消息 m 和签名 $\{r,s\}$ 发送给验证方。

3. 验证过程

验证方接收到消息 m 和签名 $\{r,s\}$ 后，按照以下步骤对签名进行验证：

(1) 计算 w：$w=s^{-1}\bmod p$，亦即 f_3；

(2) 计算 u_1：$u_1=w\cdot H(m)\bmod q$；

(3) 计算 u_2：$u_2=rw \bmod q$；

(4) 计算 v：$v=[(g^{u_1}y^{u_2})\bmod p]\bmod q$，亦即 f_4；

(5) 验证 $r=v$ 是否成立。如果成立，则签名有效；否则，签名无效。

由上述签名验证过程可以看出，待消息空间为 Z_p^*，而签名结果的空间为 $Z_p\times Z_p$。另外，DSS 标准规定 DSA 算法必须将 SHA[9,10] (secure hash algorithm) 作为哈希取值函数。

4. DSA 算法证明描述

证明：$v=[(g^{u_1}y^{u_2})\bmod p]\bmod q$

$$\Leftrightarrow v=[(g^{(H(m)w)\bmod q}y^{rw\bmod q})\bmod p]\bmod q$$

$$\Leftrightarrow v=[(g^{(H(m)w)\bmod q}g^{xrw\bmod q})\bmod p]\bmod q$$

$$\Leftrightarrow v=\{[g^{(H(m)w)\bmod q+xrw\bmod q}]\bmod p\}\bmod q$$

$$\Leftrightarrow v=\{[g^{w(H(m)+xr)\bmod q}]\bmod p\}\bmod q$$

$$\Leftrightarrow v=(g^k\bmod p)\bmod q$$

$$\Leftrightarrow v=r$$ □

3.2.7 几个特殊的数字签名

在前几节中我们讲述了一些普通的数字签名方案，但在日常应用中我们会遇到不同的情况和需求。为了满足这种需求，研究者提出了各种应用在不同情况下的特殊数字签名方案，以解决或者部分解决某些现实问题。下面介绍几个特殊用途的签名方案：盲签名、不可否认签名、代理签名和群签名。

3.2.7.1 盲签名

盲签名 (blind signature) 的概念是由 David Chaum[11] 于 1982 年提出。盲签名方案[12~15]是一个有关两个实体的密码系统，包括请求签名方和签名者。盲签名允许请求签名方能够拥有签名者所签署的消息的签名，同时签名者在签名过程中无法得到任何关于自己所签署消息的内容。也就是说，签名者只是对消息进行数字签名，而不能知道待签消息的实际内容。盲签名主要应用于数字现金、电子投票等领域[16~21]。

盲签名过程如图 3.2.3 所示。请求签名方把待签的明文消息 m 通过盲变换成为 M，从而把明文 m 的内容隐藏起来，然后把 M 发给签名者进行数字签名；签名者在签名后把签名结果 $\mathrm{Sig}(M)$ 发回给请求签名方；请求签名方把收到的签名 $\mathrm{Sig}(M)$ 进行解盲变换后即可得到签名者对消息 m 的签名 $\mathrm{Sig}(m)$。

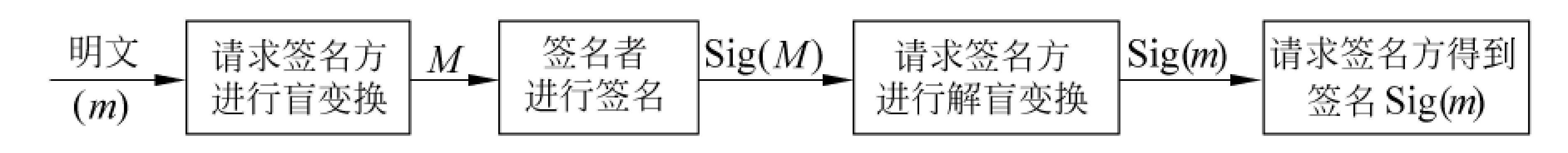


图 3.2.3　盲签名过程

3.2.7.2　不可否认签名

不可否认签名(undeniable signatures)的概念由 Chaum 和 Antwerpen[22,23] 于 1989 年提出，并且给出了一个具体的实现。与普通的数字签名一样，不可否认的数字签名，除了具有两个交互的协议，即验证协议和否认协议外，还增加一个抵赖协议(disavowal protocol)，即只有在得到签名者的许可后才能进行验证，亦即在没有签名者的合作的情况下，请求签名方将无法验证签名的合法性。不可否认签名主要由以下 3 部分组成：

(1) 签名过程：签名者 A 对消息进行数字签名，其他人不能伪造该签名；

(2) 确认过程：请求签名方 B 和签名者 A 执行交互式协议，以确认该签名的有效性；

(3) 否认协议：签名者 A 和请求签名方 B 执行的交互式协议，使得签名者 A 能够向请求签名方 B 证明某个签名不是自己签署的；不属于签名者 A 的签名一定能够通过否认协议，属于签名者 A 的合法签名(即签名者 A 进行欺骗)通过否认协议的概率极小而可以忽略。

不可否认的签名可以应用在许多方面，例如某公司 A 开发了一个软件，A 把该软件和对该软件的不可否认签名卖给 B。B 当面验证 A 的签名，以确认该软件的真实性。现在，假若 B 想把该软件的复制品私自卖给第三方 C，但由于没有公司 A 的参与，因而 C 无法验证该软件的真实性，从而保护了公司 A 的利益。不可否认签名把签名者与消息之间的关系和签名者与签名之间的关系分开。在这种签名方案中，任何人能够验证签名者实际产生的签名，验证方还需要验证该消息的签名是否有效。

但是不可否认签名也有缺点：假若签名者不愿意合作或者签名者不能被利用，签名就不能被验证。因为，不可否认数字签名只有在得到原始签名者的合作下才可以进行验证，所以，签名者可以拒绝合作或在某种情况(网络繁忙等)下不能参与合作。基于这种情况，Chaum 引进了证实数字签名[24]的概念。证实签名中引入了半可信任的第三方，他完成签名的证实和否认。当然，半可信任的第三方不能参与签名的计算，他只给签名验证者提供该签名的证实。很明显，证实签名比不可否认签名有所进步，它克服了不可否认签名的缺点，为签名的验证提供了可靠的保障。可证实签名的方案[25~28]也出现了不少，这方面的研究还在不断继续，并提供更加安全保障的方案，以满足实际应用的要求。

3.2.7.3　代理签名

代理签名(agent signature scheme)是指用户由于某种原因指定某个代理代替自己签名。该概念由 Mambo[29] 等人于 1996 年提出。例如，A 需要出差，而这些地方不能很好地访问计算机网络。因此，A 希望接收一些重要的电子邮件，并指示其秘书 B 作相应的回信。A 在不把其私钥给 B 的情况下，可以请 B 代理。

代理签名具有以下几个方面的特性[30~32]：

(1) 可区分性(distinguishability)：任何人都可以区别代理签名和正常的签名。

（2）不可伪造性（unforgeability）：只有原始签名者和指定的代理签名者能够产生有效的代理签名。

（3）代理签名的差异（deviation）：代理签名者必须创建一个能够检测到是否代理签名的有效代理签名。

（4）可验证性（verifiability）：从代理签名中，验证者能够相信原始的签名者认同了这份签名消息。

（5）可识别性（identifiability）：原始签名者能够从代理签名中识别代理签名者的身份。

（6）不可否认性（undeniability）：代理签名者不能否认由其建立且被认可的代理签名。

另外，从授权的程度上可以划分为 3 类：完全授权（full delegation）、部分授权（partial delegation）和许可授权（delegation by warrant）。

3.2.7.4 群签名

群体密码学（group-oriented cryptography）于 1987 年由 Desmedt[33] 提出。它是研究面向社团或群体中所有成员需要的密码体制。在群体密码中，有一个公用的公钥，群体外面的人可以用它向群体发送加密消息，密文收到后要由群体内部成员的子集共同进行解密。群体签名，又称团体签名（group signature）是面向群体密码学中的一个课题，1991 年由 Chaum 和 Heyst[34] 提出，具有以下几个特点：

（1）只有群中成员才能代表群体签名；

（2）接收到签名的人可以用公钥验证群签名，但不可能知道由群体中哪个成员所签；

（3）在发生争议时，可由群体中的成员或可信赖的第三方来识别该签名的签字者。

一个应用群体签名的例子：例如由投标公司组成的一个群体，一般情况下并不知道哪一份标书是属于哪一家公司签名的，而到该标书被选中之后才能识别出是哪一家公司。又如一个公司有几台计算机，每台都连在局域网上。公司的每个部门都有自己的打印机，也连在局域网上，只有本部门的人员才被允许使用他们部门的打印机。因此，打印前，必须使打印机确信用户是该部门的。同时，公司不想暴露用户的姓名。然而，如果有人在当天结束时发现打印机用得太频繁，主管者必须能够找出谁滥用了那台打印机。

群体签名可使用仲裁者：

（1）仲裁者生成一大批公开密钥/私钥密钥对，并且给群体内每个成员一个不同的唯一私钥表，在任何表中密钥都是不同的。如果群体内有 n 个成员，每个成员得到 m 个密钥对，那么总共有 $n\times m$ 个密钥对。

（2）仲裁者以随机顺序公开该群体所用的公开密钥组表，并保持各个密钥属主的秘密记录。

（3）当群体内成员想对一个文件签名时，他从自己的密钥表中随机选取一个密钥。

（4）当有人想验证签名是否属于该群体时，只需查找对应公开密钥表并验证签名即可。

（5）当争议发生时，仲裁者亦可查表得知该公钥对应于哪位成员。

这个协议的问题在于需要可信的一方，而且 m 必须足够长以避免被攻击者分析出具体是哪位成员用了哪些密钥。

群签名给该群体中的成员提供了匿名性，即验证者只能信任或者不信任签名在该群中的合法性，而不知道该成员是谁，也不能从得到的签名中分析哪几个签名属于同一个人产

生。所以，群签名对于隐藏组织中的组成结构、提供群组成员的匿名性提供了技术保障，它可以应用到电子货币的发行、政府组织结构的隐藏、匿名选举、竞标等方面[35~37]。

3.3 椭圆曲线密码体制

椭圆曲线密码体制(elliptic curve cryptosystem，ECC)是基于椭圆曲线离散对数问题的公钥密码体制，最早于 1985 年由 Miller[38] 和 Koblitz[39~41] 分别独立提出，它是利用有限域上椭圆曲线有限群代替基于离散对数问题密码体制中的有限循环群后所得到的一类密码体制。利用这一方法，可以构造公钥比特数较小的公钥密码体制。

与其他密码体制相比，椭圆曲线密码体制有以下几个优点：

(1) 处理速度快。虽然在 RSA 中可以通过选取较小公钥的方法提高公钥处理速度，即提高加密和签名验证的速度，使其在加密和数字签名的验证阶段，在速度上与 ECC 有可比性(比 ECC 稍慢一些)，但在解密和数名签名阶段，在私钥的处理速度上 ECC 则远比 RSA 和 DSA 要快得多。

(2) 占用存储空间小。不存在 RSA 计算椭圆曲线有理点群上的离散对数问题的亚指数时间算法[42]，这就意味着在达到同等安全级别的前提下，椭圆曲线密码体制只需要更小的比特数，所以 ECC 的应用[43]前景非常乐观。

(3) 带宽要求低。当对长消息进行加解密时，ECC 与 RSA 密码算法带宽要求基本相同，但在传送短消息时 ECC 带宽要求比 RSA 低得多。在带宽要求低的无线网络领域具有广泛的应用前景。

3.3.1 椭圆曲线基本概念

椭圆曲线(elliptic curve，EC)不是我们通常所指的椭圆，而是指光滑的 Weierstrass 方程所确定的平面曲线，描述如下：

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \tag{3.3.1}$$

其中，$a_i \in F$，$i=1,2,\cdots,6$，F 是一个域。F 可以是一个有理域、复数域，也可以是一个有限域 $GF(p^r)$。满足上述方程的所有点(x,y)即构成椭圆曲线。

在密码学上使用椭圆曲线的目的在于：椭圆曲线上可以提供无数的有限 Abel 群，并且这些种群的结构丰富、易于实际计算，从而可以用于构造密码算法。在实际应用中，我们通常把椭圆曲线改写成：

$$y^2 = x^3 + ax + b \tag{3.3.2}$$

的形式，并要求判别式 $\Delta = 4a^3 + 27b^2 \neq 0$。其图像如图 3.3.1 所示。

椭圆曲线密码体制在模 p(或 F_p)下定义为椭圆曲线 E：$y^2 = x^3 + ax + b$，其中 $4a^3 + 27b^2 \neq 0$；模 F_2^m 下定义为椭圆曲线 E：$y^2 + xy = x^3 + ax^2 + b$，其中 $b \neq 0$，则将此曲线称为非超奇异的(nonsuper-singular)。另外，椭圆曲线还有一个特殊的点，称为无限远点(the point at infinity)或称为零点，记为 O，它并不真正在椭圆曲线 E 上。

我们在描述椭圆曲线时，经常会用到以下几个词：

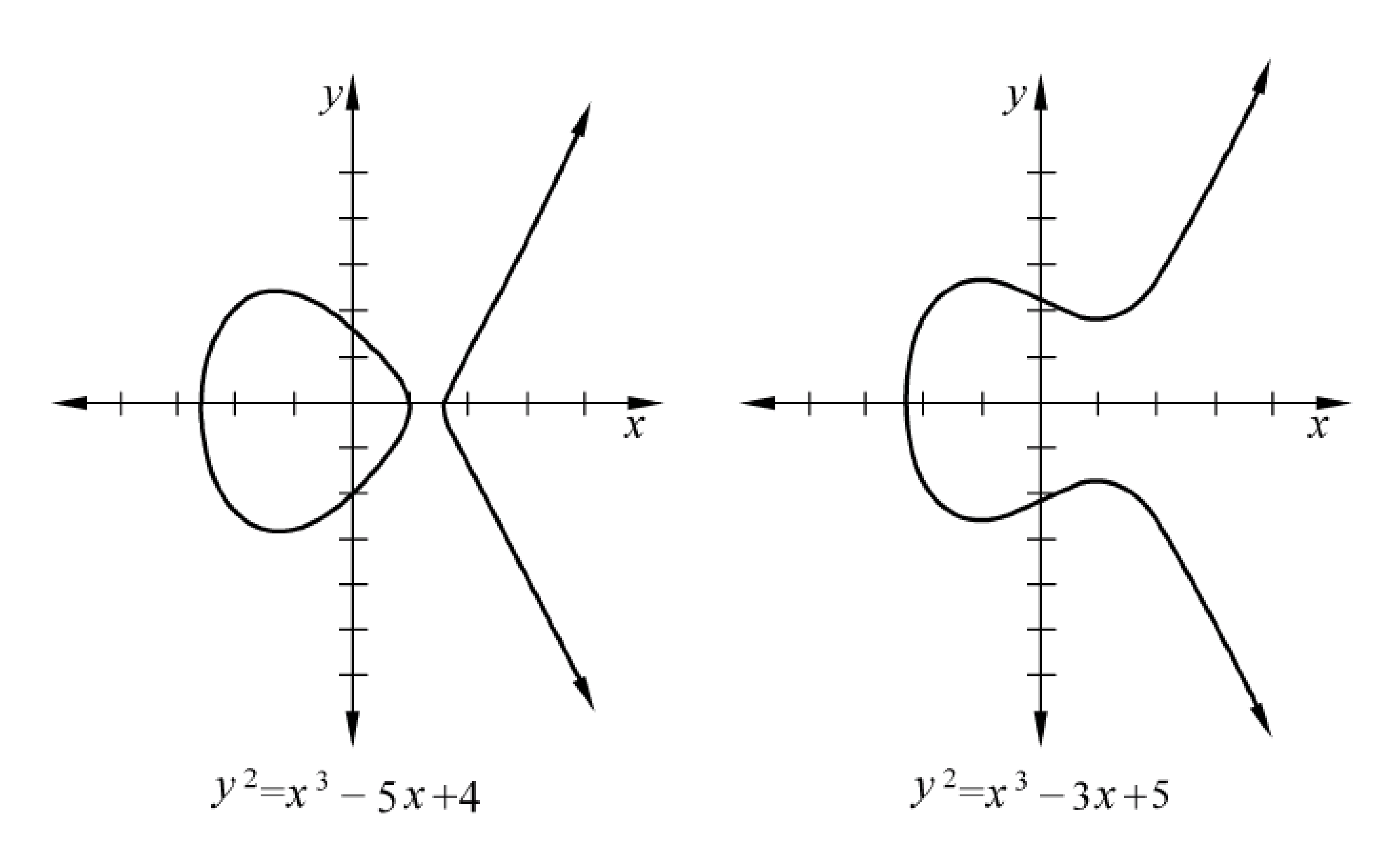


图 3.3.1 实数域中的椭圆曲线图形

(1) 负点：$E(K)$为在 K 之下椭圆曲线 E 上所有的点所构成的集合，点 $P(x,y)$对 X 坐标轴反射的点为$-P=P(x,-y)$，而称$-P$ 为点 P 的负点。

(2) 阶(order)：设 P 是椭圆曲线 E 上的一点，如果存在一个最小正整数 n，使得 $nP=O$，则称 P 点的阶为 n。定义在有限域 F_p 上的椭圆曲线上的点构成的群元素个数称为该群的阶(curve order)，用$\#E(F_p)$表示。其中，$\#E(F_p)$可以用 Hasse 定理求出。

Hasse 定理：

$$p+1-2\sqrt{p} \leqslant \# E(F_p) \leqslant p+1+2\sqrt{p} \tag{3.3.3}$$

Hasse 定理只给出了$\#E(F_p)$的取值范围，其精确值可由 Schoof[44] 或 Schoof-Elkies-Atkin(SEA)[45~47]算法求出。

(3) 基点(或称生成点)：除了无限远的点 O 之外，椭圆曲线 E 上任何可以生成有限域内全部点的点都可视为是 E 的基点 G(base point)，但并不是所有在 E 上的点都可视为基点。

3.3.2 椭圆曲线上的运算法则

3.3.2.1 椭圆曲线在模 F_p 下的运算法则

1. 加法法则

(1) 对所有的点 $P\in E(F_p)$，则 $P+O=O+P=P$，$P+(-P)=O$；

(2) 令 $P=(x_1,y_1)\in E(F_p)$及 $Q=(x_2,y_2)\in E(F_p)$，且 $P\neq -Q$，则 $P+Q=R(x_3 y_3)$，其中 $x_3=\lambda^2-x_1-x_2$，$y_3=\lambda(x_1-x_3)-y_1$；

$$\lambda=\begin{cases}\dfrac{y_2-y_1}{x_2-x_1}, & \text{如果 } P\neq Q\\[2ex] \dfrac{3x_2+a}{2y_1}, & \text{如果 } P=Q\end{cases} \tag{3.3.4}$$

(3) 如果 $s,t\in F_p$，则对所有的点 $P\in E(F_p)$而言，$(s+t)P=sP+tP$。

2. 乘法法则

(1) 如果 $k\in F_p$，则对所有的点 $P\in E(F_p)$，$kP=P+\cdots+P$(k 个 P 相加)；

(2) 如果 $s,t\in F_p$,则对所有的点 $P\in E(F_p)$,$s(t)P=(st)P$。

3.3.2.2　椭圆曲线在模 F_{2^m} 下的运算法则

1. 加法法则

(1) 对所有的点 $P\in E(F_{2^m})$,则 $P+O=O+P=P$,$P+(-P)=O$;

(2) 令 $P=(x_1,y_1)\in E(F_{2^m})$ 及 $Q=(x_2,y_2)\in E(F_{2^m})$,且 $P\neq -Q$,则 $P+Q=R(x_3y_3)$,其中,

$$x_3=\begin{cases}\left(\dfrac{y_1+y_2}{x_1+x_2}\right)^2+\dfrac{y_1+y_2}{x_1+x_2}+x_1+x_2+a, & \text{如果 } P\neq Q\\ x_1+\dfrac{b}{x_1^2}, & \text{如果 } P=Q\end{cases}\tag{3.3.5}$$

$$y_3=\begin{cases}\left(\dfrac{y_1+y_2}{x_1+x_2}\right)(x_1+x_3)+x_1+x_3, & \text{如果 } P\neq Q\\ x_1^2+\left(x_1+\dfrac{y_1}{x_1}\right)x_3+x_3, & \text{如果 } P=Q\end{cases}\tag{3.3.6}$$

(3) 如果 $s,t\in F_{2^m}$,则对所有的点 $P\in E(F_{2^m})$,$(s+t)P=sP+tP$。

2. 乘法法则

(1) 如果 $k\in F_{2^m}$,则对所有的点 $P\in E(F_{2^m})$,$kP=P+\cdots+P$(k 个 P 相加);

(2) 如果 $s,t\in F_{2^m}$,则对所有的点 $P\in E(F_{2^m})$,$s(t)P=(st)P$。

3.3.3　椭圆曲线可能遇到的攻击

椭圆曲线密码体制的安全性取决于由椭圆曲线定义的群上的离散对数问题的难度。一般的离散对数包括有效的亚指数时间算法攻击,即指数计算法(index calculus),而这种方法不能应用到椭圆曲线离散对数问题(ECDLP)上。目前,对于 ECDLP 的攻击有两类:对一般曲线的攻击和对特殊曲线的攻击。

3.3.3.1　一般椭圆曲线攻击

1. 小步大步[48,49]攻击

Shanks 提出的小步大步(baby step giant step,BSGS)攻击算法是一个基于时间存储穷举折中的算法,它不依赖于基本群的指数算法,需要群的阶的最大素因子的安全指数时间为 $O(n)$。

设 F_p 是一个阶为 n(一个大素数)的有限群,其上有一点 $P\in F_p$,并有一整数 $k\in[1,n]$。求解一个正整数 k,使得 $Q=kP$。现在令 $n_1=\sqrt{n}$,对任何 $r\in[0,n]$,r 可以唯一地表示为

$$r=an_1+b,\quad 0\leqslant a,b\leqslant n$$

这样对 m,肯定存在某 $a_0,b_0\in[0,n]$,使得

$$m=a_0n_1+b,$$

将上式代入 $Q=kP$,得到

$$Q=a_0n_1+b_0P,$$

$$[Q - a_0(n_1 P)] = b_0 P,$$

令 $P_1 = n_1 P$,得到

$$Q - a_0 P_1 = b_0 P。$$

然后,我们在$[0,n]$上采用穷举法查找出 a_0 和 b_0 的值。其步骤如下:

第 1 步(小步):依次对 $b=0,1,\cdots,n_1$ 计算 bP,并把结果列成一个表;

第 2 步(大步):依次对 $a=0,1,\cdots,n_1$ 计算 $Q-a_0P_1$,并在上表中查找;如果对某个 a 的值记为 a^*,$Q-a^*P_1$ 与表中某个 b 的值记为 b^* 所对应的 b^*P 相同,即 $Q-a^*P_1=b^*P$,则可以求出 $m=a_0n_1+b_0$。

这一算法的时间复杂度为 $O(n_1)=O(\sqrt{n})$,空间复杂度也是 $O(n_1)=O(\sqrt{n})$;因此,这一算法是对"穷搜索"方法在时间和空间上的一种折中。在用"穷搜索"方法求 m 时,其时间复杂度为 $O(\sqrt{n})$,空间复杂度为 1。

为了防止 BSGS 算法攻击,n 的长度至少应为 40 位的 10 进制数。指数计算法是一种亚指数时间算法。由于 BSGS 算法和指数算法都不能有效地用来解决椭圆曲线离散对数问题(ECDLP),这样便可以有效地防止该类攻击。

2. Pollard ρ 攻击

Pollard ρ 攻击[50,51]实际上是大步小步法的一种变形。假设有一个有限群 F_p 的阶为 n(一个大素数)。将有限域 F_p 分成 3 个大致相等的子集 S_1,S_2,S_3,然后定义 F_p 上的一个迭代函数 $f(R)(R\in F_p)$如下:

$$f(R)\begin{cases}2R, & R\in S_1\\ R+P, & R\in S_2\\ R+Q, & R\in S_3,\end{cases}$$

然后随机选取正整数 $A_0,B_0\in[1,n]$,计算迭代函数 f 的起始点 $R_0=A_0P+B_0Q$,计算,

$$R_1 = f(R_0),$$
$$R_2 = f(R_1),$$
$$\vdots$$
$$R_i = f(R_{i-1}),$$

对每一个 R_i,有

$$R_i = A_iP + B_iQ,$$

且 $A_i\leqslant A_{i+1}$,$B_i\leqslant B_{i+1}$。用(R_i,A_i,B_i)表示上式,将每次迭代中的(R_i,A_i,B_i)记录下来并将其串连成一个串。如果对某第 k_i 次迭代中的 R_k 恰好与前面第 j 次迭代中的 R_j 相同,即当 $R_k=R_j$ 时,有

$$A_iP + B_iP = A_kP + B_kP,$$

所以有

$$m \equiv \frac{A_i - A_k}{B_i - B_k} \bmod n。$$

Pollard 攻击法的时间复杂度是 $O(\sqrt{n/2})$,与大步小步法复杂度基本相同。1994 年,Oorschot 和 Wiener[52]提出了一项并行技术,这一技术使得如果使用 m 个处理器同时并行计算,其时间复杂度仅为 $O(\sqrt{n/2}/m)$。

3. Pohlig-Hellman 攻击

这种攻击法于 1978 年由 Pohlig 和 Hellman[53]提出，它实质上是一种演化算法，其基本思路是：假设 k 表示点 P 的阶，首先由 n 的素因子分解式求出 k 的素因子分解式 $k=p_1^{f_1}p_2^{f_2}\cdots p_r^{f_r}$。然后对每一个 i，$1\leqslant i\leqslant r$，求出 $m \bmod p_i^{f_i}$，最后由中国剩余定理求出 k。其中在求 $m \bmod p_i^{f_i}$ 的过程中，又将问题转化为离散对数问题 $Q'=m'P'$ 的求解问题，这里，P'的阶是 p_i，$1\leqslant m'\leqslant p_i$。

由上可知，小步大步(BSGS)方法和 Pollard 的 ρ 算法的时间复杂度基本相当，但后者的空间复杂度可以忽略，并且 Pollard ρ 算法还可以并行化处理，这对于一些特殊类型的椭圆曲线可大大地提高攻击速度，它是迄今为止所发现的最好的求解 ECDLP 的算法。因此，为了保证基于离散对数问题的密码体制的安全，所选择的群的阶必须足够大，对于有限域来说，P 的选择只有在大于 2^{160}时才能保证不受现有的各种求解离散对数问题算法的攻击。

3.3.3.2 特殊椭圆曲线攻击

特殊椭圆曲线主要有两类：一类是超奇异(super sigular)椭圆曲线，对应的攻击如 MOV 算法；另一类是异常(anomalous)椭圆曲线，其对应的攻击有 SSAS 算法。

1. MOV 攻击

MOV 算法由 Menezes，Okamoto 和 Vanstone[54]于 1991 年提出。MOV 算法利用 Weil Paring 方法，将有限域 F_p 上的椭圆曲线 E 上的 ECDLP 转化为 F_p^l 的离散对数问题(DLP)，其中 l 是满足 $p^l\gamma^l \bmod p$ 的最小正整数。而这个一般的离散对数问题可以被指数算法在亚指数时间内解决。这种算法在 $E(F_p)$是“超奇异”椭圆曲线时可以证明出 γ，而对于非“超奇异”椭圆曲线 l 的大小是不可控制的。因此为了避免 MOV 攻击，只要选择的椭圆曲线上的点 P 的秩 n 对于所有较小的 l 都不整除(p^l-1)即可。另外，其他一些 MOV 攻击见文献[55～57]。

2. SSAS 攻击

如果有限域 F_p 上的一条椭圆曲线的阶恰好为 P，则称该椭圆曲线为“异常”椭圆曲线。Semaev[58]用超越函数域的方法给出了异常 ECDLP 的攻击算法，Smart 用椭圆对数给出了另一种求异常 ECDLP 的攻击算法，Satoh 和 Araki[59]也得到了类似的结果，即 ECDLP 在一些异常椭圆曲线上是容易解的，但对于其他类型的椭圆曲线 SSAS 算法则是无效的，因此在构造椭圆曲线时只要使椭圆曲线上点的个数不等于有限域中元素个数就可以避免 SSAS 算法的攻击。

总之，对于椭圆曲线 $y^2=x^3+ax+b$，目前比较有效的攻击方法是针对椭圆曲线密码体制中的基 G 不存在一个足够大的素数和群的阶等于 P。因此，对基的选择只要注意基 G 的阶必须是一个足够大的素数和群的阶不能等于 P 即可。

3.3.4 椭圆曲线的构建

3.3.4.1 椭圆曲线域参数

椭圆曲线域参数是指构造一个椭圆曲线密码系统所需要的参数集，包括有限域GF(q)、

椭圆曲线 E、基点 G、基点的阶 n、椭圆曲线群的阶 $\#E$、相关因子 $h=\#E/n$。由于椭圆曲线可以建立于 F_p 和 F_{2^m} 两种有限域上，所以其参数的选取也有些不同。

如果选择的有限域是 F_p，则有 $q=p$，椭圆曲线方程为 $y^2=x^3+ax+b$，参数是一个六元组 (p,a,b,G,n,h)。如果选择的有限域是特征为 2 的有限域 F_{2^m}，则有 $q=2^m$，对应的椭圆曲线方程为 $y^2+xy=x^3+ax^2+b$，其参数是一个七元组 $(m,f(x),a,b,G,n,h)$（参数含义见表 3.3.1）。ANSI X9.62，IEEE P1363 和 NIST 颁布了椭圆曲线加密算法的有关标准[60,61]。

表 3.3.1 椭圆曲线参数符号

符　号	意义及说明
q	有限域的模数。当有限域为 F_p 时，$q=p$；当有限域为 F_{2^m} 时，$q=2^m$。
GF(q)	模为 q 的有限域（finite field）
E	椭圆曲线（elliptic curve）
G	基点（base point）
n	基点为 G 的阶（order）
$\#E$	椭圆曲线群的阶
h	相关因子（cofactor）$h=\frac{\#E}{n}$
a,b	椭圆曲线 E：$y^2=x^3+ax+b$（或 E：$y^2+xy=x^3+ax^2+b$）的系数
$f(x)$	有限域 F_{2^m} 中的多项式。根据 ANSI X9.62 关于选取既约多项式的建议，一般取 $f(x)=x^m+x^k+x^j+x^i+1$，其中要求 k 尽可能取小的正整数；在 k 确定后，j 尽可能取小的正整数；在 j 确定后，i 尽可能取小的正整数。

3.3.4.2 素数检测算法

素数（prime）是椭圆曲线密码体制中的重要因素，它直接影响着密码体制的安全性和基点的选取，因此正确判别一个数是否为素数是椭圆曲线中基点选择的一个重要环节。但至今还没有一种百分之百能够确定某数（特别是大奇数）为素数的算法，只有能大概确定某数为素数的算法。较著名的素数检测算法有 Solovag-Strassen 算法、Lemann 算法和 Rabin-Miller 算法。其中 Solovag-Strassen 算法的素数检测性最强，Lemann 算法次之，Rabin-Miller 算法稍差。但由于 Rabin-Miller 算法的测试时间最快，因而目前最为常用。

Rabin-Miller 算法是一种素数检测算法，描述如下：

(1) 输入待检测数 p 及一个小于 p 的随机数 a；

(2) 将 $p-1$ 表示成二进制，形成 $b_k b_{k-1}\cdots b_0$；

(3) 设定参数 d 的初始值为 1；

(4) 执行以下核心部分算法：

```
For i=k downto 0 do
      {
        x=d;
        d=d² mod P
        if (d=1) and (x≠1) and (x≠n−1) then return
```

```
        FALSE
        if b_i = 1 then d = d × a mod P
        }
    if d ≠ 1 then return FALSE
    return TRUE
```

由上述算法可知，当返回结果为 FALSE 时，p 肯定不是素数；当返回结果为 TRUE 时，则 p 有可能是素数。

3.3.4.3 椭圆曲线阶的计算

椭圆曲线的阶（$\# E$）是指椭圆曲线在有限域 $F(q)$ 上所有整数有理点的个数。最早的求椭圆曲线阶的算法是 Schoof 算法[44]，其缺点是计算性差和计算空间大。后经 Elkies[45] 和 Atkin[46] 的改进而成为现今最著名的求椭圆曲线阶的 SEA 算法。此外，还有一些 SEA 算法的变种算法[62~64]。下面就基于有限域 F_p 下对 SEA 算法进行说明。

对素数 $l(l \ll P)$，把 E 的 l-torion 点群记作 $E[l]$，则 E 的 Frobenius 映射 $\Phi:(x,y) \to (xp,yp)$ 可导出 Tate 模 $T_l(E)$ 上的线性映像，且满足方程：$\Phi^2 - t\Phi + P = 0$，其中 t 是 Frobenius 映射的迹（track）。由前述的 Hasse 定理有 $p+1-t \leqslant \# E(F_p) \leqslant p+1+t(t=2\sqrt{p})$，因此，如果找到 t_l 满足

$$\Phi^2(P) - [p](P) = t_l(P), \quad \forall P \in E[l],$$

就可以得到 $t \equiv t_l \bmod l$，选取足够多的 t_l，使 $\prod l_i > 4\sqrt{p}$，再由中国剩余定理即可算出椭圆曲线的阶 $\# E(F_p)$。

如果把第 l 可除多项式记为 $f_l(x^j)[\deg f_l = (l^2-1)/2]$，则其根恰好为 $E[l] \setminus \{O\}$ 中点的 x 坐标值。因此，$\Phi^2(P) - [p](P) = t_l(P)$ 的计算变成在环 $F_p(X,Y)/[(Y^2 = X^3 + aX + b), f_l(x)]$ 中进行，这就是 Schoof 算法，其计算复杂度为 $O(\log^8(p))$。当 $E[l]$ 上的 Frobenius 映射在 F_l 中有特征值时，称 l 为 Elkies 素数，否则称 l 为 Atkin 素数。如果 l 为 Elkies 素数，通过计算 Isogeny 映射的核（它是 Φ 的一个特征子空间）可以得到除多项式的一个因子 h_l，$\deg h_l = (l-1)/2$，然后找出适合的 λ，使得

$$\Phi(P) - [\lambda](P) \bmod (Y^2 = X^3 + aX + b), h_l(x),$$

这个 λ 即为 Φ 的特征值，由 $t \equiv \lambda + \lambda^{-1} P \bmod l$ 可以算出 $t \bmod l$。当 l 为 Atkin 素数时，Frobenius 映射的迹 $t \in F_l$ 满足 $t^2 \equiv p(\zeta + \zeta^{-1} + 2) \bmod l$，其中 $\zeta \in F_{l^2}$ 是 r 次本原单位根，从而可以计算出 $t \bmod l$ 的一个可能值 T_l。SEA 算法实际上是 Elkies 和 Atkin 两者的结合。这样，对一个素数 l，我们就有可能计算出 $t \bmod l$ 的确切值，或者可以计算出 $t \bmod l$ 的一个可能集合。再用大步小步法即可确定出 $\# E(F_p)$。

总结上述 SEA 算法，可以得出 SEA 算法的实现过程主要有两个步骤：

（1）利用 Elkies 方法、Atkin 方法、Isogeny cycles 方法以及 Virtual 方法，收集 Frobenius 映像迹 t 模一系列小素数的信息。

（2）综合所得信息，得到 t 的一个候选值集合 T，利用大步小步（BSGS）算法从候选值集合 T 中选取 t 的确切值，从而计算出曲线的阶。

此外，随着新算法的出现及对现有算法的不断优化，Morain[65~69] 等人对 SEA 算法进行了一些改进。

3.3.4.4 基点(生成点)的产生

椭圆曲线上的一点 G(一般 G 点为基点)存在一个最小正整数 n(一个素数),使得 $nG=O$,则称 n 为点 G 的阶。由上述几种对椭圆曲线的攻击方法可知,n 越大,其安全性越高。基点的产生主要有两个步骤:首先计算出候选基点,然后验证候选基点的安全性。

1. 候选基点的算法

候选基是指只是按一定的算法计算出椭圆曲线的基点,并没有对它的安全性进行验证。以下给出几种计算候选基的方法。

方法 1:由椭圆曲线方程 $E:y^2=x^3+ax+b \pmod p$ 的定义可知,只要在 E 上寻找到一个 x,使得它与 E 是平方剩余时,则可求得该 x 所对应的候选基的纵坐标值 $y_G=\sqrt{x^2+ax+b}$。但这种方法的缺点是效率比较低、计算量很大,所以只适合特定的范围数。

方法 2:由于在有限域上 $E:y^2=x^3+ax+b \pmod p$ 上的点都是整数点,我们假设 $y_{i+1}=y_i+1(i\in[1,p-1])$,则有 $y_{i+1}^2=y_i^2+2y_i+1 \pmod p$。因此,只要对上式不断作迭代计算,直到找到一个 x,使得 $y^2=x^2+ax+b \pmod p$ 存在平方剩余,则可以求得该 x 所对应的候选基的纵坐标值 $y_G=\sqrt{x^2+ax+b}$。这种方法比较适合硬件实现,其计算效率还是比较低。

方法 3:假设在椭圆曲线上一点 $G'(m,n)$,将这一点代入 $E:y^2=x^3+ax+b \pmod p$ 可以得到等式 $n^2=m^2+am+b \pmod p$,并改写成 $b=n^2-m^2+am \pmod p$。从这个等式中不难发现 b 由 m,n 和 a 决定,也就是说,如果将随机选取 b 改为随机选取 m 和 n,再令 $b=n^2-m^2+am \pmod p$。这样就可以在选取椭圆曲线的同时也选定了候选基点。这是一种随机的选取方法。由前面定理可知,只要保证椭圆曲线的群阶为素数,就可以保证基点选择的任意性和安全性。

总之,方法 1 在不超过现有编译软件所定义的整数范围内是一个相当好的计算候选基的算法,并且这种可能是很大的,因为每一个整数存在平方剩余的概率为一半,而现有的编译软件有 64 位模长,也就是说,在这个范围内不存在椭圆曲线上的点的概率几乎小到 0。方法 2 用硬件并行实现效率最好,如果用软件实现,这种算法不如方法 1。方法 3 在椭圆曲线阶为素数的情况下是最好的方法。

2. 候选基点的安全性检验

由椭圆曲线阶($\#E$)的性质可知,只要 $\#E$ 为大素数就可以确保所选择的候选基点是安全的,即我们只要对 $\#E$ 进行素数测试即可。基于这一思想,可按如下步骤进行安全性测试:

(1) 用 SEA 算法计算椭圆曲线的阶 $\#E$,检查 $\#E$ 是否小于安全需要。如果不能满足要求,则重新选择 p,a,b,并继续执行第(1)步;否则执行(2);

(2) 检查 $\#E$ 是否为素数,如果为真,则输出 G 点为基,否则执行(3);

(3) 对 $\#E$ 进行素数因子分解为 $n_1,n_2,\cdots,n_i$,其中 n_i 为素数,i 为小于 1000 的正整数。检查 $n_1,n_2,\cdots,n_i$ 是否都小于安全要求,如果为真,则重新选择 p,a,b 并跳回第(1)步,否则继续执行(4);

(4) 对 $n_1,n_2,\cdots,n_i$ 进行排序,生成一个互不相同的序列 $n_1^i,n_2^i,\cdots,n_m^i$,其中 $m\leqslant i$;

(5) 分别计算 $n_1^iG,n_2^iG,\cdots,n_m^iG$ 的值;

(6) 如果存在一个 $n_t^i G=O$，其中 $1\leqslant t\leqslant m$，检测 n_t^i 是否满足安全需要。当 n_t^i 不满足要求或者不存在 $n_t^i G=O$ 时，需要重新选择基点，再重新执行第(5)步；或者重新选择 p,a,b，跳回第(1)步；如果存在一个素数 $n_t^i G=O$，但 n_t^i 又比用户要求的阶大，则同样重新选择基点后再重新执行第(5)步，直到满足用户要求为止；

(7) 最后把 G 点输出并作为基。

3. 基点产生的全过程

综合上述步骤，基点 $G(x_G, y_G)$ 的产生全过程描述如下：

(1) 随机选择正整数 $a,p,m,n,a,m,n\in[1,p]$。

(2) 检测 p 是否为素数且 p 不小于所要求的域；如果不是，则转到(1)。

(3) 计算 $b\equiv n^2-m^2+am(\bmod p)$。

(4) 检查 $4a^3+27b^2=0(\bmod p)$；如果为真，则转到(1)。

(5) 利用 SEA 算法计算椭圆曲线的阶 $\#E$。

(6) 用前述的素数检测 $\#E$ 是否为素数；如果 $\#E$ 不为素数，或者 $\#E=p$，又或者小于安全需要，则转到(1)。

(7) 输出基点 $(x_G, y_G)=(m,n)$。

3.3.4.5 安全椭圆曲线的产生

下面我们将分别介绍基于有限域 F_p 和 F_{2^m} 的椭圆曲线产生过程。

1. 产生基于有限域 F_p 的椭圆曲线方程

基于有限域 F_p 的椭圆曲线 E：$y^2=x^3+ax+b(\bmod p)$，$a,b\in F_p$，$\Delta=4a^3+27b^2\neq 0(\bmod p)$。由于 ANSI X9.62 推荐 p 为不少于 160 位的素数；为了使每次产生的椭圆曲线方程都不一样，引入一个随机比特串 Random_Seed，同时再引入一个单向哈希函数 SHA-1。另外，在 p 确定的情况下，我们设 $t=\lceil \log p \rceil$，$s=[(t-1)/160]$，$v=t-160s$。其中，t 可以理解为 p 的二进制串的长度或长度减 1，s 为 p 有多少个整 160 位长度，v 为 p 的串长度除以 160 后的余量。

下面给出产生椭圆曲线的算法：

(1) 输入一个大于 160 位的素数 p；

(2) 随机产生一个长度为 g 位的比特串 Random_Seed，$g>160$；

(3) 计算 $H=\text{SHA-1}(\text{Random_Seed})$，并把 H 的最右边 v 位赋给 c_0；

(4) 把 c_0 最左边一位(第 v 位)设置为 0，把修改后 c_0 的值赋给 $W_0(W_0\leftarrow c_0)$，以确保 $r<p$；

(5) 把 g 位的 Random_Seed 转换成整数，并记为 z；

(6) 执行以下循环：

```
for i =1 to s
{
  s_i =(z+i)mod 2^g ;
  W_i =SHA-1(s_i)             /* 把 s_i 作为比特串看待 */
  }
```

(7) 把 $W_0, W_1, \cdots, W_{s-1}, W_s$ 串联成 W：$W = W_0 \parallel W_1 \parallel \cdots \parallel W_{s-1} \parallel W_s$；

(8) 把 W 转换成整数 r；

(9) 如果 $r=0$ 或 $4r+27=0 \bmod p$，则跳回步骤(1)；

(10) 随机选择 $a,b,a,b \in F_p$ 且 a,b 不同时为 0，并要求 $rb^2 = a^3 \bmod p$；

(11) 输出(Random_Seed，a，b)。

2. 产生基于有限域 F_{2^m} 的椭圆曲线方程

基于有限域 $F_q(q=2^m)$ 的椭圆曲线 E：$y^2+xy=x^3+ax^2+b(\bmod q)$，$a,b \in F_q$。与基于有限域 F_p 的椭圆曲线类似，我们选取 m 为不少于 160 位的素数，并引入一个随机比特串 Random_Seed 和一个单向哈希函数 SHA-1。而在 q 确定的情况下，设 $s=[(m-1)/160]$，$v=m-160s$。其中，t 表示为 m 的二进制串的长度，s 为 q 有多少个整 160 位长度，v 为 q 的串长度除以 160 后的余量。

下面给出产生椭圆曲线的算法：

(1) 输入一个大于 160 位的素数 m；

(2) 随机产生一个长度为 g 位的比特串 Random_Seed，$g>160$；

(3) 计算 $H=$SHA-1(Random_Seed)，并把 H 的最右边 v 位赋给 b_0；

(4) 把 g 位的 Random_Seed 转换成整数，并记为 z；

(5) 执行以下循环：

```
for i =1 to s
{
  s_i =(z+i)mod 2^g;
  b_i =SHA-1(s_i) /*  把 s_i 作为比特串看待 */
  }
```

(6) 把 $b_0, b_1, \cdots, b_{s-1}, b_s$ 串联成 b：$b=b_0 \parallel b_1 \parallel \cdots \parallel b_{s-1} \parallel b_s$；

(7) 如果 $b=0$，则跳回步骤(1)；

(8) 随机选择一个 a；

(9) 输出(Random_Seed，a，b)。

最后说明一点，随机产生在 F_p 上的椭圆曲线算法中的 r 是与群同态的概念有关；同样，随机产生在 F_{2^m} 上的椭圆曲线算法中的 b 也是与群同态的概念有关。

3.3.5 基于椭圆曲线的密码体制

对椭圆曲线及其算法有了基本了解后，本节将对基于有限域求解离散对数问题的 Diffie-Hellman 密钥交换、ElGamal 密码体制和 DSA 密码体制，分别用椭圆曲线来实现其密码体制。

3.3.5.1 基于椭圆曲线的 Diffie-Hellman 密钥交换

图 3.3.2 给出了 Diffie-Hellman 密钥交换过程及算法验证。首先用前述的方法把椭圆

曲线及各参数产生，包括椭圆曲线方程 $E_q(a,b)$、基点 G、基点的阶 n 等，然后把 $E_q(a,b)$ 和 G 公开。

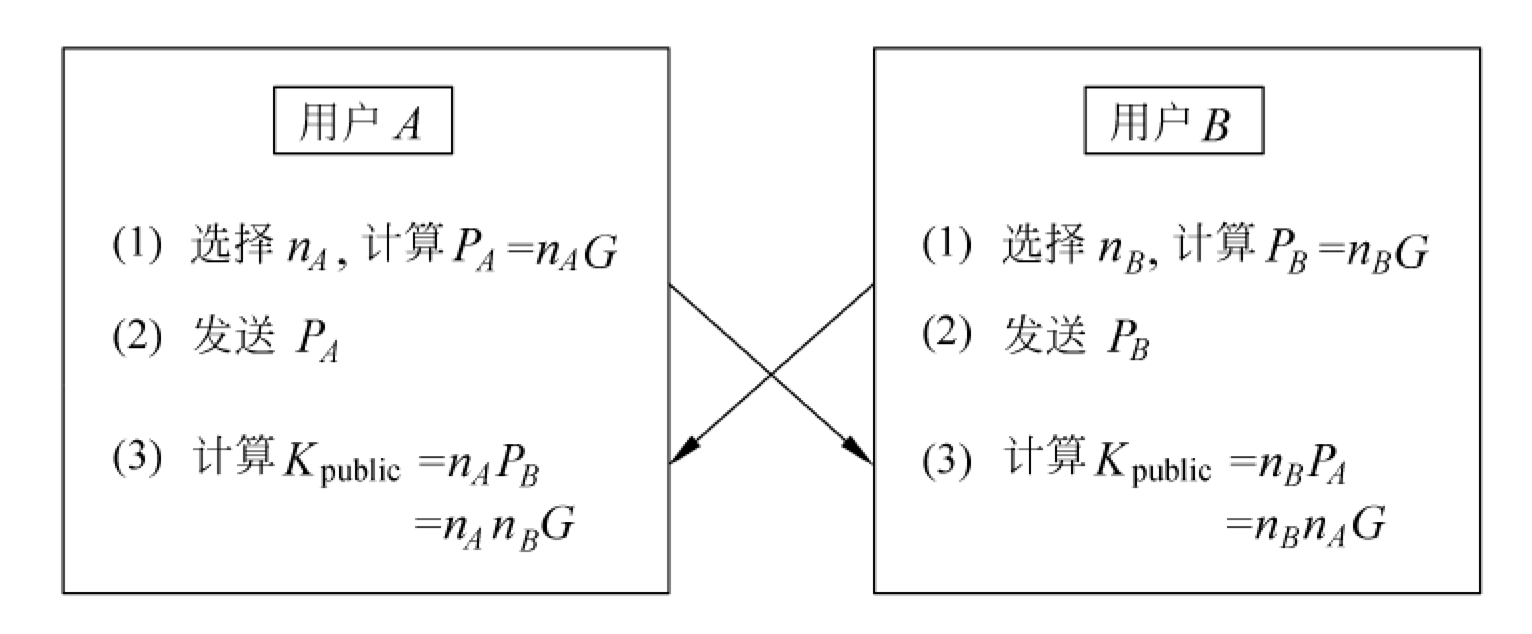


图 3.3.2　基于椭圆曲线的 Diffie-Hellman 密钥交换

用户 A 和 B 之间的密钥交换过程描述如下：

(1) A 选择一个小于 n 的整数 n_A($n_A\in[1,n-1]$)作为其私钥，计算 $P_A=n_AG$ 并把 P_A 作为其公钥；

(2) 同样地，B 选择一个小于 n 的整数 n_B($n_B\in[1,n-1]$)作为其私钥，计算 $P_B=n_BG$ 并把 P_B 作为其公钥；

(3) A 和 B 分别由 $K_{public}=n_AP_B$ 和 $K_{public}=n_BP_A$ 产生双方的共享密钥 K_{public}。

算法验证：

用户 A：$K_{public}=n_AP_B=n_An_BG$

用户 B：$K_{public}=n_BP_A=n_An_BG$

3.3.5.2　基于椭圆曲线的 ElGamal 密码算法

与 Diffie-Hellman 密钥交换的实现类似，首先生成椭圆曲线方程 $E_q(a,b)$，基点 G，基点的阶 n 等，然后把 $E_q(a,b)$ 和 G 公开。

图 3.3.3 为基于椭圆曲线的 ElGamal 加/解密过程，其加/解密过程描述如下。

1. 参数初始化

(1) 将待加密信息 m 通过编码嵌入到椭圆曲线上的一点 P_m；

(2) 用户 A 随机选择一个整数 n_A($n_A\in[1,n-1]$)；

(3) 计算 $P_A=n_AG$ 并把 P_A 作为其公钥公布。

2. 加密过程

(1) 若用户 B 要向 A 发送消息 P_m，则随机选择一个整数 n_B($n_B\in[1,n-1]$)；

(2) 计算点 $P_1=n_BG$ 和 $P_2=P_m+n_BP_A$；

(3) 将密文 $M=\{P_1,P_2\}$发送给 A。

3. 解密过程

(1) 用户 A 接收到密文 M 后，作如下运算：$P_m=P_2-n_AP_1$，即可得到 P_m；

(2) 对经过编码处理的点 P_m 进行译码，从而得到明文 m。

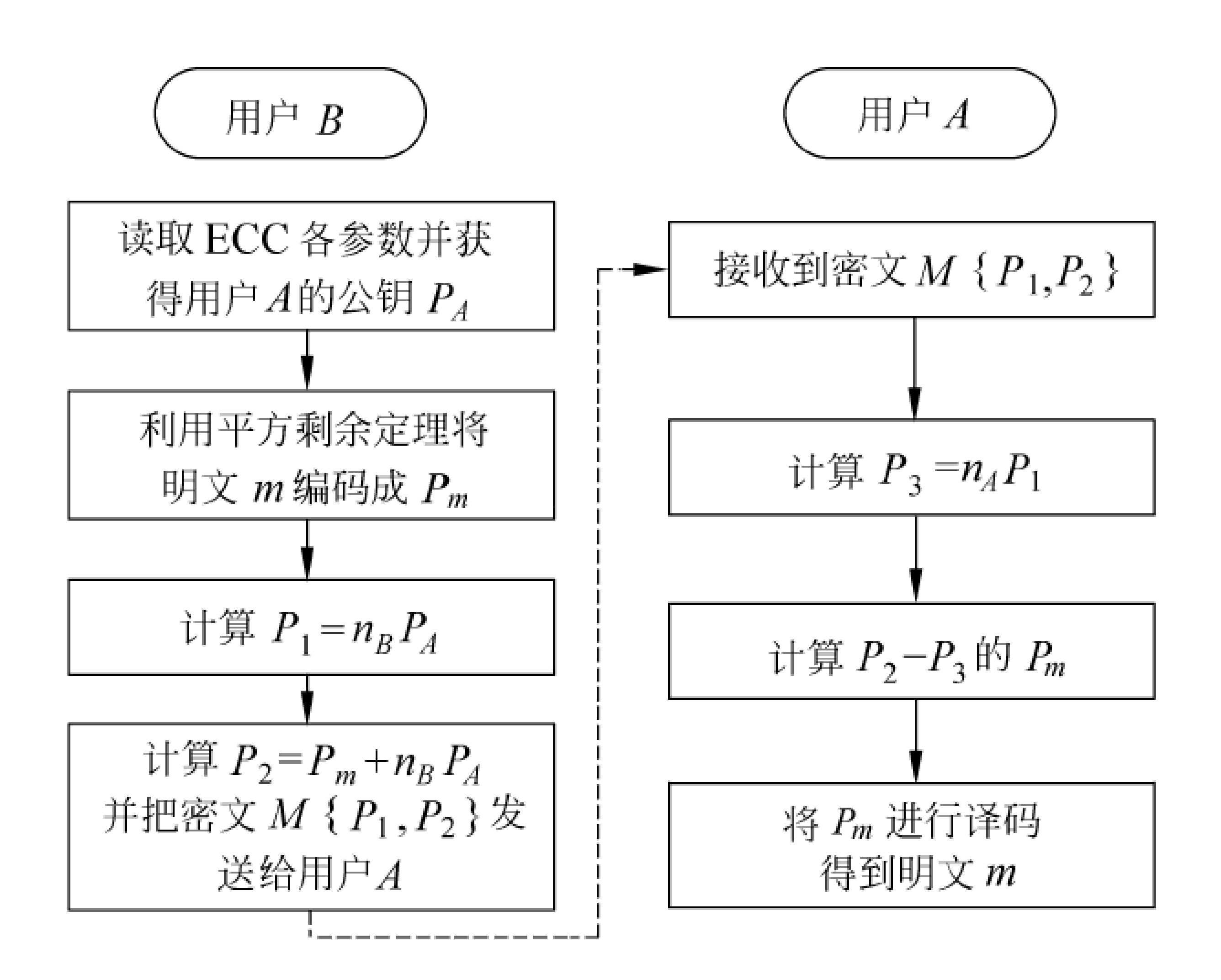


图 3.3.3　基于椭圆曲线的 ElGamal 加(解)密过程

4. 算法证明

$P_2-P_1=(P_m+n_BP_A)-n_A(n_BG)=P_m+n_B(n_AG)-n_A(n_BG)=P$。 □

3.3.5.3 基于椭圆曲线的 DSA(ECDSA)签名算法

同样地,首先按上节介绍的方法产生曲线方程及其他相关参数,下面对基于椭圆曲线的 DSA(ECDSA)的数字签名进行描述,如图 3.3.4 所示。

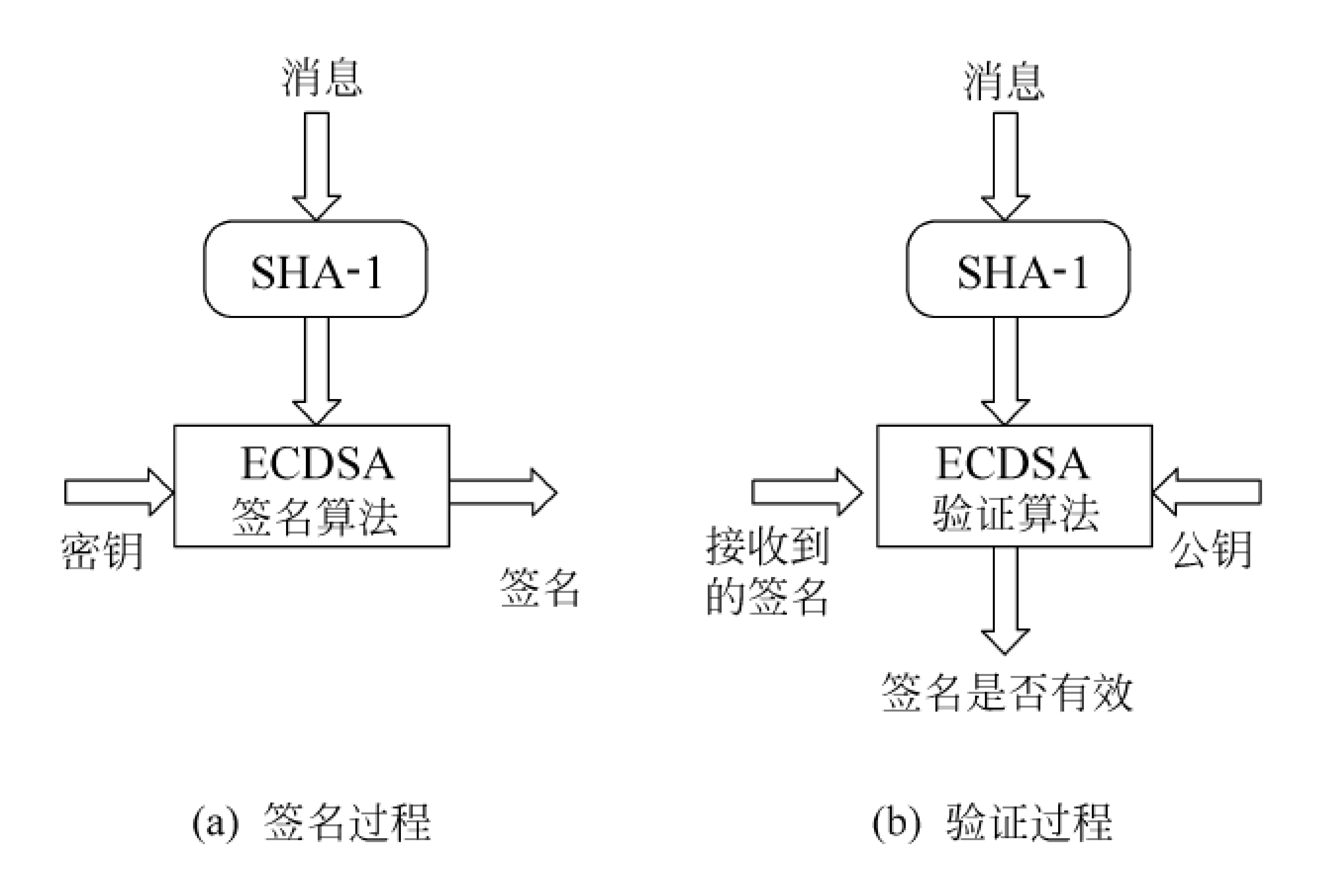


图 3.3.4　Diffie-Hellman 密钥交换

假设用户 A 要对消息 m 签名,由用户 B 验证其签名的有效性。

1. 参数初始化

(1) 选择一个大素数 p,$p>2^{160}$;

(2) 选择一条椭圆曲线方程 E: $y^2=x^3+ax+b$,$4a^3+27b^2\neq 0 \bmod p$;

(3) 确定一个有限域 F_p;

(4) 确定椭圆曲线的序 $\#E$,$p+1-2\sqrt{p}\leqslant \#E(F_p)\leqslant p+1+2\sqrt{p}$;

(5) 计算生成点 G，其序为 n；

(6) 选定一个单向哈希函数 SHA-1；

(7) 选取一个 n_A，$n_A \in [1, n-1]$作为私钥；

(8) 计算公钥 $Q = n_A G$；

(9) 公布参数 p, E, G, n, F_p，SHA-1 和 Q。

2. 签名过程

(1) 选择一个 k，$k \in [1, n-1]$；

(2) 计算 $kG = (x_1, y_1)$；

(3) 计算 $r = x_1 \bmod n$；如果 $r = 0$，则返回步骤(1)；

(4) 计算 $k^{-1} \bmod n$；

(5) 计算 $e = \text{SHA-1}(m)$；

(6) 计算 $s = k^{-1}(e + n_A r) \bmod n$；如果 $s = 0$，则返回步骤(1)；

(7) 产生对消息 m 的签名$\{r, s\}$；

(8) 把$\{m, r, s\}$发送给用户 B。

3. 认证过程

(1) 检查 $r, s \in [1, n-1]$是否成立；如果成立，则继续下一步骤，否则，签名无效；

(2) 计算 $e = \text{SHA-1}(m)$；

(3) 计算 $w = s^{-1} \bmod n$；

(4) 计算 $u_1 = ew \bmod n$，$u_2 = rw \bmod n$；

(5) 计算点 $X(x_1, y_1) = u_1 G + u_2 Q$，如果 $X = O$，则签名无效；

(6) 计算 $v = x_1 \bmod n$；

(7) 如果 $v = r$，则签名有效，否则无效。

4. 算法验证

由签名过程可知，如果签名$\{r, s\}$是消息 m 的合法签名，则有 $s = k^{-1}(e + n_A r) \bmod n$。故此，我们要证明上述等式是否成立。

证明： $s = k^{-1}(e + n_A r) \bmod n$

$$\Leftrightarrow k = s^{-1}(e + n_A r) \bmod n$$

$$\Leftrightarrow k = (s^{-1} e + s^{-1} n_A r) \bmod n$$

$$\Leftrightarrow k = (we + wn_A r) \bmod n \qquad (\text{由 } w = s^{-1} \bmod n)$$

$$\Leftrightarrow k = (u_1 + u_2 n_A) \bmod n \qquad (\text{由 } u_1 = rw \bmod n, u_2 = rw \bmod n)$$

$$\Leftrightarrow kG = (u_1 + u_2 n_A) G \bmod n$$

$$\Leftrightarrow kG = u_1 G + u_2 Q \bmod n \qquad (\text{由 } Q = n_A G)$$

因为 kG 的横坐标 $x_1 = r$；$u_1 G + u_2 Q$ 的横坐标为 $x_1 = v$

所以有 $v = r$。 □

3.3.6 椭圆曲线的性能及安全性分析

3.3.6.1 性能分析

公钥密码体制的有效性主要考虑 3 个因素：①计算开销；②密钥长度；③通信传输量（带宽）。在对不同密码系统之间进行有效性比较时，应基于相同的安全等级。下面我们将就上述 3 个因素对椭圆曲线密码体制的性能与其他密码系统进行比较。

（1）计算开销

计算开销是指变换公钥和私钥所需的计算量。对于 RSA 和基于 ECC 的椭圆曲线数字签名算法(ECDSA)或椭圆曲线加密方案(ECES)，大部分数字签名和加密变换操作都可以进行预计算。在相同的测试环境下，在 163 位 ECC/1024 位 RSA 安全级数下，ECC-163 的计算开销比 RSA-1024 大 5～15 倍。在 256 位 ECC/3072 位 RSA 安全级数下，ECC-256 与 RSA-3072 的计算开销比率已增加到 20～60 倍[70~72]。而 ECC-521 则比 RSA-15360 快 400 倍。表 3.3.2 给出了在相同测试环境下 ECC 与 RSA 计算开销的比较。

表 3.3.2　相同测试环境下 ECC 与 RSA 计算开销的比较　　毫秒

CPU/MHz	ECC-163 位	ECC-192 位	RSA-1024-d	RSA-1024-e
450	6.1	8.7	32.1	1.7
400	22.9	37.7	188.7	10.8

（2）存储空间

密钥长度决定存储密钥对和系统参数所需要的比特数。与 RSA/DSA 相比，ECC 只需使用更小的系统参数即可达到同等的安全级数[73]（见表 3.3.3），其密钥存储空间非常小。

表 3.3.3　ECC、DSA 和 RSA 系统参数和密钥长度的比较

算法	系统参数/bits	公钥/bits	私钥/bits
ECC	481	161	160
DSA	2208	1024	160
RSA	—	1088	2048

（3）通信传输量

通信传输量是指发送一条加密消息或一条签名信息所需传输的比特数。在公钥密码系统中，双方在进行通信之前，必须先进行密钥交换或对长消息进行数字签名，此过程是短消息传输的。实验证明，虽然在传输长信息时 ECC 的传输量与 RSA 的传输量相当，但在传送短消息时，ECC 的通信传输量却比 RSA 低很多，见表 3.3.4。

表 3.3.4　ECC、DSA 和 RSA 3 种算法签名长度和加密后的密文长度的比较

算法	签名长度/bits	密文大小/bits
ECC	320	321
DSA	320	2048
RSA	1024	1024

3.3.6.2 安全性分析

虽然椭圆曲线点运算的概念是很容易理解的，但在实现椭圆曲线密码时有许多关键性的问题需要解决，其中主要包括两个方面：一是如何选取合适的符合安全条件的随机椭圆曲线；二是如何快速实现椭圆曲线密码。合适的椭圆曲线参数一旦产生就可以形成椭圆曲线群，能够提供给多个用户使用，生成其公钥、私钥对。加密算法的安全性能一般通过该算法的抗攻击强度来反映。ECC 和其他几种公钥系统相比，其抗攻击性具有绝对的优势。以当前应用较为广泛的公钥系统 RSA 为例，RSA 方法的优点主要在于原理简单，易于使用。但是，随着整数因子分解方法的不断完善、计算机速度的提高以及计算机网络的发展，作为 RSA 加解密安全保障的大整数要求越来越大。要保证 RSA 使用的安全性，就要相应地增加其密钥的位数，目前一般认为 RSA 需要 1024 位以上的字长才有安全保障。但是，密钥长度的增加导致了其加解密的速度大为降低，硬件实现也变得越来越困难，这对使用 RSA 的应用带来了很重的负担，从而使得其应用范围越来越受到制约。而椭圆曲线则具有较短的密钥长度，例如 160 位 ECC 与 1024 位 RSA、DSA 具有相同的安全强度，210 位 ECC 则与 2048 位的 RSA、DSA 具有相同的安全强度，这就意味着 ECC 对带宽的要求更低，所占有的存储空间更小。

现有已知求解 ECDLP 的著名算法有：穷举搜索法、大步小步算法、Pollard's Rho 算法、并行 Pollard's Rho 算法等，它们的难度都是指数级的，见表 3.3.5。而以往基于模运算的大整数因式分解问题（IFP）和离散对数问题（DLP）都只是存在亚指数时间复杂度的通用算法。正是由于 ECC 算法所存在这一明显不同，使得 ECC 算法的单位安全强度高于其他（如 RSA）算法，也就是说，在达到同样安全强度的情况下，ECC 算法所需的密钥长度远比 RSA 算法要低。并且，随着安全级数的提高，RSA 密钥长度的增长速度远远快于 ECC 密钥长度的增长，见表 3.3.6。

表 3.3.5　各种 ECDLP 算法与 RSA 算法时间复杂度比较

算　法	时间复杂度
穷举搜索法	$O(n)$
Pollard's Rho 算法	$O(\pi 2/n)$
并行 Pollard's Rho 算法	$O(\sqrt{\pi n}/2m)$
大步小步算法	$O(\sqrt{n})$
RSA	$O(\exp(\sqrt{(\ln p)\ln(\ln p)}))$

表 3.3.6　相同安全级别下 ECC 与 RSA 密钥长度比较

ECC 密钥长度	RSA 密钥长度	ECC 与 RSA 密钥长度比
106	512	1∶5
163	1024	1∶6
233	2048	1∶9
283	3072	1∶11
409	7680	1∶19
517	15 360	1∶27

通过以上分析可以看出，在相同的安全条件下，ECC 的密钥尺寸要远远小于 RSA/DSA，同时，随着密钥长度的增加，ECC 的安全性要比 RSA/DSA 的安全性增加快得多，因此 ECC 具有更高的安全强度。随着在有限域上的离散对数问题和因子分解上不断取得进展，为了达到安全要求，大多数公钥密码体制的密钥也越来越大。椭圆曲线密码体制相对于其他公钥密码体制具有密钥长度短、运算速度快、计算数据量小等特点，因而 ECC 已成为已知的效率最高的公钥密码系统。

3.4 基于 ECC 的群体导向(t,n)门限签名方案

群体签名，又称为团体签名(group signature)，是面向群体密码学中的一个研究内容，相对其他特殊用途的数字签名而言，群签名的概念是密码学中较新的概念，到目前为止，对它的研究已经有十几年的历史了。在此期间，群数字签名在不断地完善，不断地向实用的方向进步。本节将首先对群体签名和群体(t,n)门限签名进行较详细的说明，然后对 Harn[74] 于 1994 年提出的基于离散对数的群体导向(t,n)门限数字签名方案进行阐述，最后提出一个改善了 Harn 方案思想、基于椭圆曲线密码体制的群体导向(t,n)门限数字签名方案。

3.4.1 群体签名与(t,n)门限签名

1. 群签名的定义

群签名方案的概念由 Chaum 与 Heyst[34] 于 1991 年首先在“Group Signatures”一文中提出。他们在文章中给出了群签名所必须具有的性质：

(1) 任何成员都能够代表群进行签名；

(2) 签名的接收者可以验证签名是否为该群的成员所产生的合法签名，但是接收者不能从签名中恢复原始签名者的身份，也无法判断两个群签名是否是由同一个群成员提交的；

(3) 在产生纠纷的情况下，被指定的可信密钥认证中心(trusted key authentication center，KAC)能够打开群签名，以便揭示签名者的身份。

显然，群签名方案提供了数字签名的匿名性，这种特性对某些实际应用是必需的。Chaum 与 Heyst 所提出的方案及其在文中阐述的内容，给群体签名的研究提供了许多启发，如从群签名建立的数学基础、群签名的组成部分、群签名的运行效率(如公钥的大小、计算量和通信量)等方面对群签名进行研究与发展，以后 Lee，Chang，Tseng 和 Jan 等人[75~78] 提出了各种新的群体签名方案。

群签名概念的提出，使得数字签名的应用前景进一步拓宽，为网络信息安全提供新的技术支持。

2. 群签名的安全性和效率要求

一个好的群签名方案应满足如下安全性要求：

(1) 不可伪造性(unforgeability)：只有群成员能够代表群进行签名；

(2) 匿名性(anonymous)：给定一个群签名，除 KAC 之外，任何人想识别签名者的身份

都是计算困难的；

(3) 不关联性(unlinkability)：除 KAC 之外，判断两个不同的群签名是否是由同一个群成员提交是困难的；

(4) 不可替代性(exculpability)：任何一个群成员与 KAC 都不能代表其他群成员进行签名；

(5) 可跟踪性(traceability)：KAC 能够打开一个有效的群签名，以揭示签名者的身份，并且任何签名者都不能阻碍一个有效群签名的打开；

(6) 抗联合攻击(coalition-resistance)：群中一部分成员串通在一起也不能产生一个合法的不能被跟踪的群签名[79]。

而一个群签名方案的效率主要依赖于以下几个方面：

(1) 群公钥的大小；

(2) 群签名的长度；

(3) 群签名算法和验证算法的效率；

(4) 创建算法，注册协议以及打开算法的效率。

综合上述对群体签名的介绍，高效、实用的群签名方案应该满足如下要求：

(1) 群组的公开钥的大小不依赖于群组成员的多少；

(2) 群签名的长度不依赖于群组成员的多少；

(3) 群组中成员的变化不用重新建立系统，或者不用重新给各成员颁布新的签名密钥和群组的公开钥；

(4) 签名的大小要尽可能地小，签名的运算、签名的验证尽可能简单；

(5) KAC 与群成员之间的通信成本要小。

3. 门限签名

1988 年，Desmedt 在其发表的文章中提出了(t,n)门限签名[80]的概念。门限签名[81~83]是建立在基于群体签名的基础上，但与群签名不同的是，其主要思想是基于秘密共享[84,85]。设有 n 个人参与一个秘密共享方案，我们把秘密分成 n 部分，每部分称为它的影子(shadow)或子密钥，分发给每个参与成员一个与其他人不同的影子，它们中的任何 t 部分都能够用来重构秘密，而任意不足 t 个参与者的子密钥凑在一起不能确定秘密，亦即不能产生正确的群体签名。通常称这种形式的签名为(t,n)门限签名，其中 t 就是所谓的门限值。

(t,n)门限签名具有如下 5 种属性：

(1) 群内不少于 t 位成员合作才能产生群体签名；

(2) 群体签名的大小和验证时间与单个子密钥的大小和验证时间一样；

(3) 签名的验证变得很简单，因为只需要一个群体公钥即可完成；

(4) 群体签名可以被任何一位群体外的成员验证；

(5) 群体负责对信息进行签名。

从门限签名的特性容易看到，如果门限签名中的 t 个(或者以上)成员被攻破，那么门限签名的安全性荡然无存，所以形象地称 t 为门限值。另外，按子密钥发放方式的不同，可将门限群签名方案分成带可信密钥认证中心(KAC)的门限群签名方案与分布式门限群签名(即没有 KAC 的协助)[86~88]方案两类。

一个好的门限群签名方案应该具有如下8个特性：

（1）群特性：只有群体的成员才能作自己的部分签名，其他人无法伪造部分签名；

（2）门限特性：只有作部分签名的人数超过门限值时，门限群签名才会产生；

（3）验证简单性：签名的验证者验证签名时只需知道群体的公钥；

（4）匿名性：签名的验证者无法知道哪些成员作了哪部分签名；

（5）可追查性：事后可以追查出哪些成员作了部分签名；

（6）不可冒充性：任何成员集合不能冒充另一个成员集合作门限群签名；

（7）强壮性：恶意成员达到或超过门限值仍无法知道系统的秘密参数；

（8）稳定性：删除或加入成员时，系统参数不需要作大的修改。

门限签名的实现方法使得签名所使用的秘密参数通过秘密分享算法分配给多个人保管，需要的时候有多个人一起进行签名。这在某种程度上减小了秘密值泄露的可能性，但是最终还是不能彻底杜绝泄露的发生。于是，如何减小秘密泄露后的损失以及如何设计一个好的门限群签名方案是一个很值得研究的问题。

3.4.2 Harn(t,n)门限数字签名方案

Desmedt 和 Frankel[89]提出基于 RSA 体制的群体导向签名法，而后于1994，Harn 对 ElGamal 密码体制进行修改，把 Lagrange 内插多项式方法与 ElGamal 安全机制相结合，提出了一个群体导向(t,n)门限数字签名方案，系统的安全则建立在解离散对数问题(DLP)的困难度上[74]。

Harn 针对(t,n)群体导向门限数字签名方案提出了两种方法，第1种方法是假设系统存在一个可信密钥认证中心(KAC)，专门负责选择系统参数及计算，并分配个别密钥给每一位参与者；第2种方法则取消了 KAC 的协助，采用自我认证的方法[90,91]。

3.4.2.1 需要 KAC 帮助的 Harn 方案

Harn 所提出的这种方法，是利用 Shamir[92]的秘密共享和 Lagrange 插值多项式以及 NIST[93]中所提出的数字签名算法实现的，其方案的实现过程如下：

1. 参数初始化

首先由 KAC 选定如下参数：

（1）选择一个大素数 p 作为模，且 $2^{511}<p<2^{512}$；

（2）选择一个可以被 $p-1$ 整除的素数 q，且 $2^{159}<q<2^{160}$；

（3）随机选择一个整数 z_i，$0<z_i<q-1$ 和一条多项式：

$$f(x)=a_0+a_1x+\cdots+a_{t-1}x^{t-1}(\bmod q),\quad i=0,1,\cdots,t-1;$$

（4）在有限域 F_p 中选择一个阶为 q 的生成数 α，$\alpha=h^{(p-1)/q}\bmod p$；其中 h，$0\leqslant h\leqslant p-1$ 为一个随机整数，所以有 $h^{(p-1)/q}\bmod p>1$。

2. 群体和群体成员公/私钥产生过程

（1）由 KAC 选定群体私钥为 $f(0)$；

（2）KAC 选定每位群体成员的私钥为 $f(x_i)\bmod q$，$i=1,2,\cdots,n$；

(3) 由 KAC 计算出群体公钥 $y=\alpha^{f(0)} \bmod p$；

(4) 为了用于数字签名的验证，KAC 还需要计算出每位群体成员的个人公钥 $y_i=\alpha^{f(x_i)} \bmod p$。

3. (t,n)门限签名产生过程

假设群体中有任意 $u_1,u_2,\cdots,u_t$ 位成员要对消息 m 签名，则签名过程描述如下：

(1) 群体中每位成员 $u_i,i=1,2,\cdots,t$ 随机选择一个整数 $k_i[1,q-1]$，计算个人公钥 $r_i=\alpha^{k_i} \bmod p$，并通过广播信道把 r_i 公开；

(2) 当某位成员把所有的 r_i 都收好后，计算：

$$r=\prod_{i=1}^{t} r_i \bmod p \tag{3.4.1}$$

(3) 成员 u_i 使用其个人私钥 $f(x_i)$和 k_i 计算出对消息 m 进行签名$\{r_i,s_i\}$。其中 s_i，$0\leqslant s_i\leqslant q-1$ 为整数，其值由下述方程式确定：

$$s_i=f(x_i)\cdot f(m)\cdot\left(\prod_{j=1,j\neq i}^{t}\frac{-x_j}{x_i-x_j}\right)-k_i\cdot r \bmod p \tag{3.4.2}$$

最后，u_i 把签名$\{r_i,s_i\}$发送给指定的办事员(clerk)；

(4) 办事员在收到所有群体成员的签名$\{r_i,s_i\}$后，通过下式：

$$y_i^{f(m)\prod_{j=1,j\neq i}^{t}\frac{-x_j}{x_i-x_j}}=r_i\alpha^{s_i} \bmod p \tag{3.4.3}$$

对每位成员的个人签名$\{r_i,s_i\}$的真伪性进行验证。如果所有的签名$\{r_i,s_i\}$都为真，则计算：

$$s=\sum_{i=1}^{t} s_i \bmod p \tag{3.4.4}$$

最后得出群体签名$\{r,s\}$。

4. (t,n)门限签名验证过程

外部人员在收到该群体签名$\{r,s\}$和消息 m 后，只需要用下式对签名的正确性进行验证：

$$y^{f(m)}=r^r\alpha^s \bmod p \tag{3.4.5}$$

如果上式成立，则说明签名$\{r,s\}$有效。

3.4.2.2　不需要 KAC 帮助的 Harn 方案

由于在这种方案中没有大家所信任的 KAC 帮助产生所需要的参数，因而群体中的所有成员需要一起协商来产生其个人私钥，并把其私钥发布给群体内的其他成员。

以下这些参数必须得到群体内所有成员的一致认同，包括：

(1) 一个大素数 p 作为模，且 $2^{511}<p<2^{512}$；

(2) 素数 q，它可以被 $p-1$ 整除，且 $2^{159}<q<2^{160}$；

(3) 在有限域 F_p 中选择一个阶为 q 的生成数 α，$\alpha=h^{(p-1)/q} \bmod p$；其中 h，$0\leqslant h\leqslant p-1$ 为一个随机整数，所以有 $h^{(p-1)/q}(\bmod p)>1$。

1. 公钥产生过程

(1) 群体内每位成员随机选择一个整数 z_i 和 x_i，其中 $z_i,x_i\in[1,p-1]$；并把 z_i 作为个

人密钥；

(2) 计算 $y_i = \alpha^{z_i} \bmod p$；

(3) 结合 x_i 和 y_i 得出每位成员的个人公钥为$\{x_i, y_i\}$；

(4) 计算出群体公钥 y：

$$y = \prod_{i=1}^{n} y_i \bmod p \tag{3.4.6}$$

由于没有 KAC 帮助，所以每位成员除了产生个人密钥和公钥外，还需要把自己看作一个 KAC，操作过程如前述的$(t, n-1)$秘密共享方案一样，并把个人的密钥发布给其他的 $n-1$位成员。现在假设其中有某一成员 u_i(其私钥为 z_i)，u_i 随机选择一条$(t-1)$阶多项式 $f_i(x)$，并有 $f_i(0)=z_i \bmod p$；计算其他 $n-1$ 位成员，如 $u_j(i \neq j)$的个人密钥 $f_i(x_j) \bmod q$ 和个人公钥 $y_{i,j} = \alpha^{f_i(x_j)} \bmod p$。

2. (t, n)门限签名产生过程

假设群体(t, n)中有任意 $u_1, u_2, \cdots, u_t$ 位成员要对消息 m 签名，并且各自的签名过程可以同时进行。下面对某位成员 u_i 的签名过程进行描述：

(1) 成员随机选择一个整数 $k_i[1, q-1]$，并计算个人公钥 $r_i = \alpha^{k_i} \bmod p$，通过广播信道把 r_i 公开；

(2) 当每位成员把所有的 r_i 都收好后，计算：

$$r = \prod_{i=1}^{t} r_i \bmod p \tag{3.4.7}$$

(3) 成员 u_i 使用其个人私钥 z_i, k_i 和 $f_j(x_i)(j=t+1, t+2, \cdots, n)$，对消息 m 进行签名$\{r_i, s_i\}$。其中 s_i，$0 \leqslant s_i \leqslant q-1$ 为整数，其值由下述方程式确定：

$$s_i = \{z_i + [\sum_{j=t+1}^{n} f_j(x_i)] \cdot \left(\prod_{k=1, k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right) \cdot f(m) - k_i \cdot r\} (\bmod q) \tag{3.4.8}$$

最后，u_i 把签名$\{r_i, s_i\}$发送给指定的办事员；

(4) 当办事员收到由成员 u_i 发送过来的签名$\{r_i, s_i\}$和信息 m 后，利用 u_i 的公钥 x_i, y_i 和 $y_{j,i}(j=t+1, t+2, \cdots, n)$，通过下式：

$$\left\{y_i \left(\sum_{j=t+1}^{n} y_{j,i}\right)^{\prod_{k=1, k\neq i}^{t} \frac{-x_k}{x_i - x_k}}\right\}^{f(m)} = r_i^r \alpha^{s_i} (\bmod p) \tag{3.4.9}$$

来验证其签名的有效性。如果上式成立，则成员 u_i 的签名$\{r_i, s_i\}$真实有效；

(5) 当办事员把 t 位成员的签名$\{r_i, s_i\}$都收齐并验证其真实性后，办事员利用 u_i 的公钥 x_i, y_i 和 $y_{j,i}(j=t+1, t+2, \cdots, n)$，通过下式：

$$s = \sum_{i=1}^{t} s_i \bmod q \tag{3.4.10}$$

得出群体签名$\{r, s\}$。

3. (t, n)门限签名验证过程

外部人员在收到该群体签名$\{r, s\}$和消息 m 后，只需要利用下式对签名的正确性进行验证：

$$y^{f(m)} = r^r \alpha^s \bmod p \tag{3.4.11}$$

如果上式成立，则说明群体签名$\{r,s\}$有效。

3.4.3　基于椭圆曲线密码体制的(t,n)门限数字签名方案

本节将就上述 Harn 所提出的方案，在分别基于存在 KAC 协助和没有 KAC 协助两种情况下，结合 Chen[94] 方法，利用椭圆曲线密码体制分别将其实现。与 Harn 方案一样，我们假设有一个群体 $U=\{u_1,u_2,\cdots,u_n\}$，共有 n 位成员。根据群体导向(t,n)门限数字签名方案的定义，群体中最少有 t 位成员共同签名才能作有效的群体签名。下面将就上述这两种情况进行实现描述。

3.4.3.1　需要 KAC 帮助的(t,n)门限数字签名方案

1. 参数初始化

首先由 KAC 负责产生以下各参数：

(1) 选择一个大素数 p，$p>2^{160}$；

(2) 选择一条椭圆曲线方程 E：$y^2=x^3+ax+b$，$4a^3+27b^2\neq 0 \pmod p$；

(3) 确定一个有限域 F_p，并且 F_p 的所有点(x,y)满足方程 E；$E(F_p)$表示为 $E\cup\{O\}$，其中 O 为无穷远点；

(4) 确定椭圆曲线的阶 n，$p+1-2\sqrt{p}\leqslant n\leqslant p+1+2\sqrt{p}$；

(5) 计算生成点 $G\in E(F_p)$，其阶为 n；

(6) 选定一个单向哈希函数 $h()$；

(7) 选取一条秘密的多项式 $f(x)=a_{t-1}x^{t-1}+a_{t-2}x^{t-2}+\cdots+a_1x+a_0 \pmod n$，$a_i\in[1,n-1]$，$i=0,1,\cdots,t-1$；

(8) 给群体中所有成员分配个人私钥 $f(x_i)$，并计算其个人公钥 $y_i=f(x_i)\cdot G$；其中，x_i 为用户 $u_i(i=1,2,\cdots,n)$的公开身份识别码；

(9) 令群体私钥为 $f(0)=a_0$，并计算群体公钥 $y=f(0)\cdot G$；

(10) KAC 把上述参数中的 $p,E,F_p,n,G,h(),y,x_i,y_i(i=1,2,\cdots,n)$公开。

2. (t,n)门限签名产生过程

假设在群体 $U=\{u_1,u_2,\cdots,u_n\}$中有 t 位成员想合作对消息 m 进行签名，可依照如下步骤共同产生群体签名。

(1) 每位参与签名的成员 u_i 选取一个随机数 $k_i\in[1,n-1]$，并计算 $r_i=k_i\cdot G$，同时把 r_i 广播给其他成员；

(2) 当每位参与者接收了所有的 r_i 后，计算：

$$r=\left(\sum_{j=1}^{t}r_j\right)\pmod p \tag{3.4.12}$$

和

$$s_i=k_i\cdot X_r-f(x_i)\cdot h(m)\cdot\prod_{j=1,j\neq i}^{t}\frac{-x_i}{x_i-x_j}\pmod n \tag{3.4.13}$$

于是得出对消息 m 的个人签名$\{r_i,s_i\}$，并把签名$\{r_i,s_i\}$和信息 m 发送给办事员。上式中的 X_r 代表点 r 在 x 坐标轴上的值；

(3) 当办事员收到由所有参与签名的成员 u_i 发送过来的签名 $\{r_i, s_i\}$ 和信息 m 后，通过下式判别签名的真伪：

$$r_i \cdot X_r = s_i \cdot G + y_i \cdot h(m) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j} (\bmod n) \tag{3.4.14}$$

如果上式成立，则代表来自成员 u_i 的签名 $\{r_i, s_i\}$ 是真实无误的。

3. (t, n) 门限签名验证过程

当办事员分别将 t 位参与成员的个人签名收到并确定为真实后，计算：

$$s = \left(\sum_{j=1}^{t} s_j\right) (\bmod n) \tag{3.4.15}$$

最后得出群体签名 $\{r, s\}$。

另外，任何一位参与签名的成员只要利用群体公钥 y，并通过下面的等式就可以验证群体签名 $\{r, s\}$ 的真伪性：

$$r \cdot X_r = s \cdot G + h(m) \cdot y (\bmod p) \tag{3.4.16}$$

若上式成立，则代表该群体签名是正确的，否则 $\{r, s\}$ 是无效的。

4. 定理证明

定理 3.4.1 判别式 $r_i \cdot X_r = s_i \cdot G + y_i \cdot h(m) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}$ 成立。

证明： 把式(3.4.13)两边同时乘以生成点 G 得：

$$s_i \cdot G = \left\{k_i \cdot X_r - f(x_i) \cdot h(m) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}\right\} \cdot G$$

$$\Leftrightarrow s_i \cdot G = (k_i \cdot G) \cdot X_r - [f(x_i) \cdot G] \cdot h(m) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}$$

$$\Leftrightarrow s_i \cdot G = r_i \cdot X_r - y_i \cdot h(m) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}$$

$$\Leftrightarrow r_i \cdot X_r = s_i \cdot G + y_i \cdot h(m) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}$$

□

定理 3.4.2 判别式 $r \cdot X_r = s \cdot G + h(m) \cdot y$ 成立。

证明： 把定理 3.4.1 中结论两边同取 $\sum$，有：

$$\sum_{i=1}^{t} [r_i \cdot X_r] = \sum_{i=1}^{t} \left[s_i \cdot G + y_i \cdot h(m) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}\right]$$

$$\Leftrightarrow X_r \cdot r = s \cdot G + h(m) \cdot \sum_{i=1}^{t} \left[f(x_i) \cdot G \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}\right]$$

$$\Leftrightarrow X_r \cdot \sum_{i=1}^{t} r_i = \left(\sum_{i=1}^{t} s_i\right) \cdot G + h(m) \cdot \sum_{i=1}^{t} \left[y_i \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}\right]$$

$$\Leftrightarrow X_r \cdot r = s \cdot G + h(m) \cdot G \cdot \sum_{i=1}^{t} \left[f(x_i) \cdot \prod_{j=1, j\neq i}^{t} \frac{-x_j}{x_i - x_j}\right] \tag{3.4.17}$$

由 Lagrange 插值公式得知：$f(x) = \sum_{i=1}^{t} \left[f(x_i) \cdot \prod_{j=1, j\neq i}^{t} \frac{x - x_j}{x_i - x_j}\right]$

于是我们有：$f(0)=\sum_{i=1}^{t}\left[f(x_i)\cdot\prod_{j=1,j\neq i}^{t}\frac{0-x_j}{x_i-x_j}\right]=a_0$

所以，式(3.4.17)改写为

$$X_r\cdot r=s\cdot G+h(m)\cdot G\cdot f(0)$$
$$\Leftrightarrow r\cdot X_r=s\cdot G+h(m)\cdot y$$ □

3.4.3.2　不需要 KAC 帮助的(t,n)门限数字签名方案

由于没有 KAC 的帮助，在群体公钥和私钥的产生过程中，可以将其看成(n,n)签名方案的形式。而群体内所有成员必须自行协调产生并认同以下各参数：

(1) 选择一个大素数 p，$p>2^{160}$；

(2) 选择一条椭圆曲线方程 E：$y^2=x^3+ax+b$，$4a^3+27b^2\neq 0 \pmod p$；

(3) 确定一个有限域 F_p，并且 F_p 为所有点(x,y)满足方程 E；$E(F_p)$表示为 $E\cup\{O\}$，其中 O 为无穷远点；

(4) 确定椭圆曲线的阶 n，$p+1-2\sqrt{p}\leqslant n\leqslant p+1+2\sqrt{p}$，并且 $n>2^{160}$；

(5) 计算生成点 $G\in E(F_p)$，其阶为 n；

(6) 选定一个单向哈希函数 $h()$；

(7) 选取一条秘密的多项式 $f(x)=a_{t-1}x^{t-1}+a_{1t-2}x^{t-2}+\cdots+a_1x+a_0 \pmod n$，$a_i\in[1,n-1]$，$i=0,1,\cdots,t-1$。

1. 公钥产生过程

(1) 所有群体成员随机选取一个整数 $z_i\in[1,n-1]$，把 z_i 作为其个人私钥，并且使 $f_i(0)=z_i \pmod n$；随机选取一个 $x_i\in[1,n-1]$作为其公开身份认证码；

(2) 计算相关的公钥 $y_i=z_i\cdot G$；

(3) 结合 x_i 和 y_i 得出每位成员的个人公钥为$\{x_i,y_i\}$，$\{z_i\}$作为私钥；

(4) 计算出群体公钥 y：

$$y=\prod_{i=1}^{n}y_i \bmod p \tag{3.4.18}$$

由于没有 KAC 的帮助，所以每位成员除了产生个人密钥和公钥外，还需要把自己看作一个 KAC，操作过程如前述的$(t,n-1)$秘密共享方案一样，并把个人的密钥发布给其他 $n-1$ 位成员。现在假设其中有某一成员 u_i(其私钥为 z_i)，u_i 随机选择一条$(t-1)$阶多项式 $f_i(x)$；计算其他 $n-1$ 位成员，如 $u_j(i\neq j)$的个人密钥 $f_i(x_j)$和个人公钥 $y_{i,j}=f_i(x_j)\cdot G$。

2. (t,n)门限签名产生过程

假设在群体 $U=\{u_1,u_2,\cdots,u_n\}$中有 t 位成员想合作对消息 m 进行签名，可依照如下步骤共同产生群体签名：

(1) 每位参与签名的成员 $u_i(i=1,2,\cdots,t)$，选取一个随机数 $k_i\in[1,n-1]$，并计算 $r_i=k_i\cdot G$，同时把 r_i 广播给其他成员；

(2) 当每位参与签名的成员收到所有的 r_i 后，计算：

$$r=\left(\sum_{i=1}^{t}r_i\right) \pmod p \tag{3.4.19}$$

和

$$s_i = \left\{z_i + \left[\sum_{j=t+1}^{n} f_j(x_i)\right] \cdot \left(\prod_{k=1,k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right) \cdot h(m) - k_i \cdot X_r\right\} (\bmod n) \qquad (3.4.20)$$

于是得出对消息 m 的个人签名$\{r_i, s_i\}$，并把签名$\{r_i, s_i\}$和信息 m 发送给办事员。其中，X_r 代表点 r 在 x 坐标轴上的值；

（3）当办事员收到由所有参与成员 u_i 发送过来的签名$\{r_i, s_i\}$和信息 m 后，办事员利用 u_i 的个人参数 x_i, y_i 和 $y_{i,j}$，$j=t+1, t+2, \cdots, n$，通过下式判别签名的真伪：

$$h(m) \cdot y_i + h(m) \cdot G \cdot \left[\sum_{j=t+1}^{n} f_j(x_i) \cdot \prod_{k=1,k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right] = s_i \cdot G + r_i \cdot X_r (\bmod n)$$

（3.4.21）

如果上式成立，则代表来自成员 u_i 对信息 m 的个人签名$\{r_i, s_i\}$是真实无误的。

3. (t, n)门限签名验证过程

当办事员分别将 t 位参与成员的个人签名收到并确定为真实后，计算：

$$s = \left(\sum_{i=1}^{t} s_i\right) (\bmod n) \qquad (3.4.22)$$

最后得出群体签名$\{r, s\}$。

另外，任意一位参与签名的成员只要利用群体公钥 y，并通过下面的等式来验证群体签名$\{r, s\}$的真伪性：

$$h(m) \cdot y = s \cdot G + r \cdot X_r (\bmod p) \qquad (3.4.23)$$

若上式成立，则代表该群体签名是正确的，否则$\{r, s\}$是无效的。

4. 定理证明

定理 3.4.3 判别式：

$$h(m) \cdot y_i + h(m) \cdot G \cdot \left[\sum_{j=t+1}^{n} f_j(x_i) \cdot \prod_{k=1,k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right] = s_i \cdot G + r_i \cdot X_r (\bmod n)$$

成立。

证明：把式(3.4.20)左右两边同时乘以生成点 G，得到：

$$s_i \cdot G = \left\{z_i + \left[\sum_{j=t+1}^{n} f_j(x_i)\right] \cdot \left(\prod_{k=1,k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right) \cdot h(m) - k_i \cdot X_r\right\} \cdot G$$

$$\Leftrightarrow s_i \cdot G + k_i \cdot G \cdot X_r = h(m) \cdot \left\{z_i \cdot G + G \cdot \left[\left(\sum_{j=t+1}^{n} f_j(x_i)\right) \cdot \prod_{k=1,k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right]\right\}$$

$$\Leftrightarrow s_i \cdot G + r_i \cdot X_r = h(m) \cdot \left\{y_i + G \cdot \left[\left(\sum_{j=t+1}^{n} f_j(x_i)\right) \cdot \prod_{k=1,k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right]\right\}$$

$$\Leftrightarrow s_i \cdot G + r_i \cdot X_r = h(m) \cdot y_i + h(m) \cdot G \cdot \left\{\left[\left(\sum_{j=t+1}^{n} f_j(x_i)\right) \cdot \prod_{k=1,k\neq i}^{t} \frac{-x_k}{x_i - x_k}\right]\right\}$$

□

定理 3.4.4 判别式 $h(m) \cdot y = s \cdot G + r \cdot X_r (\bmod p)$成立。

证明：把定理 3.4.3 中结论两边同取 $\sum$，得：

$$\sum_{i=1}^{t} [s_i \cdot G + r_i \cdot X_r]$$

$$= \sum_{i=1}^{t}\left\{h(m)\cdot y_i + h(m)\cdot G\cdot\left[\left(\sum_{j=t+1}^{n} f_j(x_i)\right)\cdot \prod_{k=1,k\neq i}^{t}\frac{-x_k}{x_i - x_k}\right]\right\}$$

$$\Leftrightarrow G\cdot\sum_{i=1}^{t} s_i + X_r\cdot\sum_{i=1}^{t} r_i$$

$$= h(m)\cdot\sum_{i=1}^{t} y_i + h(m)\cdot G\cdot\sum_{i=1}^{t}\left\{\left[\sum_{j=t+1}^{n} f_j(x_i)\right]\cdot\prod_{k=1,k\neq i}^{t}\frac{-x_k}{x_i - x_k}\right\} \tag{3.4.24}$$

由 Lagrange 插值公式及假设条件得知：

$$\left[\sum_{j=t+1}^{n} f_j(x_i)\right]\cdot\prod_{k=1,k\neq i}^{t}\frac{-x_k}{x_i - x_k} = f_j(0) = z_j。$$

所以式(3.4.24)改写为：

$$\Leftrightarrow s\cdot G + r\cdot X_r = h(m)\cdot\sum_{i=1}^{t} y_i + h(m)\cdot G\cdot\sum_{j=i+1}^{n} z_j$$

$$\Leftrightarrow s\cdot G + r\cdot X_r = h(m)\cdot\sum_{i=1}^{t} y_i + h(m)\cdot\sum_{j=i+1}^{n} z_j\cdot G$$

$$\Leftrightarrow s\cdot G + r\cdot X_r = h(m)\cdot\sum_{i=1}^{t} y_i + h(m)\cdot\sum_{j=i+1}^{n} y_j$$

$$\Leftrightarrow s\cdot G + r\cdot X_r = h(m)\cdot\left[\sum_{i=1}^{t} y_i + \sum_{j=i+1}^{n} y_j\right]$$

$$\Leftrightarrow s\cdot G + r\cdot X_r = h(m)\cdot\sum_{i=1}^{n} y_i$$

$$\Leftrightarrow s\cdot G + r\cdot X_r = h(m)\cdot y$$ □

3.4.3.3　安全性分析

本方案的安全性基于椭圆曲线离散对数问题(ECDLP)的安全性之上。下面将就本方案可能遭受到的几种攻击加以说明：

(1) 在有 KAC 存在的情况下，若攻击者企图由群体公钥 $y=f(0)\cdot G$ 中求得群体私钥 $f(0)$，则必须面对解椭圆曲线离散对数问题(ECDLP)的困难，其难度远大于解离散对数问题，在计算上可以说是不可能实现的。

(2) 在有 KAC 存在的情况下，若攻击者企图由个人公钥 $y_i=f(x_i)\cdot G$ 中求得个人私钥 $f(x_i)$，同样要面对求椭圆曲线离散对数问题(ECDLP)的困难。若攻击者企图由多项式方程组：

$$s_i = k_i\cdot X_r - f(x_i)\cdot h(m)\cdot\prod_{j=1,j\neq i}^{t}\frac{-x_i}{x_i - x_j}\pmod p$$

解得个人私钥 $f(x_i)$，则先求得 k_i，而求解 k_i 同样是要面对解椭圆曲线离散对数问题(ECDLP)的困难。

(3) 在有 KAC 存在的情况下，假设攻击者想根据判别方程：

$$r_i\cdot X_r = s_i\cdot G + y_i\cdot h(m)\cdot\prod_{j=1,j\neq i}^{t}\frac{-x_j}{x_i - x_j}$$

伪造某一参与成员 u_i 对消息 m 的个人签名$\{r_i',s_i'\}$。攻击者首先会随机选择一个整数 $k_i\in[1,n-1]$，计算 $r'=k_i'\cdot G$，并把 r' 广播出去给其他成员。但由于攻击者无法通过求解

ECDLP 而求出成员 u_i 的个人私钥 $f(x_i)$，从而无法求解出满足上述方程的 s_i'值。

(4) 在没有 KAC 协助的情况下，若攻击者想冒充签名，其在求解判别式

$$h(m)\cdot y_i+h(m)\cdot G\cdot\left[\sum_{j=t+1}^{n}f_j(x_i)\cdot\prod_{k=1,k\neq i}^{t}\frac{-x_k}{x_i-x_k}\right]=s_i\cdot G+r_i\cdot X_r(\bmod n)$$

时同样要面对求解 ECDLP 的问题。

(5) 若群体内某攻击者在获得参数 $h(m),G,s,r$ 和 y 等参数后，想伪造另一个群体签名$\{r',s'\}$，同样要面对解椭圆曲线离散对数问题(ECDLP)的困难。

3.4.3.4 性能分析

本性能分析主要从计算开销和通信传输量两个方面对 Harn 方案与本文提出的方案进行比较分析。前者的分析可作为系统计算效能提升的说明，后者则说明系统运作时的通信带宽，使系统的性能评估更为具体。

1. 计算开销

在进行计算开销比较分析之前，首先定义如下符号，见表 3.4.1。

表 3.4.1 符号定义

符 号	意义及说明	符 号	意义及说明
T_{H}	执行单向哈希函数所需的时间	T_{INV}	模逆元素运算所需的时间
T_{MUL}	模乘法运算所需的时间	$T_{\mathrm{EC_MUL}}$	椭圆曲线乘法运算所需的时间
T_{DIV}	模除法运算所需的时间	$T_{\mathrm{EC_ADD}}$	椭圆曲线加法运算所需的时间
T_{EXP}	模指数运算所需的时间		

根据文献[95]中的假设得知：

(1) 对于模指数 $g^k \bmod p$ 的运算，p 是一个 1024 位的大质数，k 是一个 160 位随机整数。

(2) 对于椭圆曲线乘法运算，计算 kG 是对于所有的点 $P\in E(F_p)$ 且 $p\approx 2^{160}$，其中 k 是一个 160 位随机整数。

由此可以推得不同运算量与模乘法运算量的相对关系如下：

$$T_{\mathrm{EXP}}\approx 240T_{\mathrm{MUL}};\quad T_{\mathrm{EC_MUL}}\approx 29T_{\mathrm{MUL}};\quad T_{\mathrm{EC_ADD}}\approx 0.12T_{\mathrm{MUL}}。$$

而根据文献[96]可知，$T_{\mathrm{DIV}}\approx T_{\mathrm{MUL}}$，$T_{\mathrm{INV}}\approx[0.843\ln(q)+1.47]T_{\mathrm{MUL}}$。此外，相对于模数乘法，模数加法和模数减法的运算量相当低，且运算也快，对系统运作时的执行效率影响较小，因此其运算量可忽略不计。

Harn 方案与本方案计算开销的比较见表 3.4.2 和表 3.4.3。

表 3.4.2 Harn 方案与本方案的计算开销比较(有 KAC 的协助)

阶 段	Harn 方案	本方案
(t,n)门限签名产生过程	$2(t+1)T_{\mathrm{EXP}}+(5t+1)T_{\mathrm{MUL}}+2tT_{\mathrm{H}}$ $\approx(485t+481)T_{\mathrm{MUL}}+2tT_{\mathrm{H}}$	$(3t+3)T_{\mathrm{ECC_MUL}}+(t+1)T_{\mathrm{ECC_ADD}}$ $+T_{\mathrm{MUL}}+(t+1)T_{\mathrm{H}}$ $\approx(87.12t+88.12)T_{\mathrm{MUL}}+(t+1)T_{\mathrm{H}}$
(t,n)门限签名验证过程	$3T_{\mathrm{EXP}}+T_{\mathrm{MUL}}+T_{\mathrm{H}}\approx 721T_{\mathrm{MUL}}+T_{\mathrm{H}}$	$3T_{\mathrm{EC_MUL}}+(t+1)T_{\mathrm{EC_ADD}}+T_{\mathrm{H}}$ $\approx(0.12t+87.12)T_{\mathrm{MUL}}+T_{\mathrm{H}}$

表 3.4.3 Harn 方案与本方案的计算开销比较(没有 KAC 的协助)

阶 段	Harn 方案	本方案
(t,n)门限签名产生过程	$2(t+1)T_{EXP}+(t^2+t+2)T_{MUL}+tT_H$ $\approx(t^2+481t+242)T_{MUL}+tT_H$	$(2t+5)T_{ECC_MUL}+2(t+1)T_{ECC_ADD}$ $t(t+2)T_{MUL}+(t+2)T_H$ $\approx(t^2+60.24t+145.24)T_{MUL}+(t+2)T_H$
(t,n)门限签名验证过程	$4T_{EXP}+(3t^2+5t)T_{MUL}+2tT_H$ $\approx(3t^2+965t)T_{MUL}+2tT_H$	$3T_{EC_MUL}+(t+1)T_{EC_ADD}+T_H$ $\approx(0.12t+87.12)T_{MUL}+T_H$

2. 通信传输量

在 Harn 方案中,p 的长度$|p|$为 512 位,q 的长度$|q|$为 160 位。而在我们提出的基于椭圆曲线密码体制方案中,p 和 n 的长度$|p|$和$|n|$皆为 160 位。另外,需要指出的是,椭圆曲线传送一个点坐标 $P(x,y)$所需要的传输量为 $2p$,也就是 160 位。Harn 方案与本方案的通信传输量的比较见表 3.4.4 和表 3.4.5。

表 3.4.4 Harn 方案与本方案的通信传输量比较(有 KAC 的协助)

阶 段	Harn 方案	本方案
(t,n)门限签名产生过程	$2\|p\|+\|q\|$	$2\|n\|$
(t,n)门限签名验证过程	$\|q\|$	$\|n\|$

表 3.4.5 Harn 方案与本方案的通信传输量比较(没有 KAC 的协助)

阶 段	Harn 方案	本方案
(t,n)门限签名产生过程	$2\|p\|+\|q\|$	$3\|n\|$
(t,n)门限签名验证过程	$\|q\|$	$\|n\|$

总之,本节提出的方法是以椭圆曲线密码体制为平台架构,密钥长度只需 160 位即可等同于 RSA 使用 1024 位密钥长度的安全性。在不降低其安全性的前提下,密钥长度更短,因此可以使信息在网络上的传输速度更快,所需的内存空间也得以大幅度降低。相比原来的 Harn 方法,其在安全性和效能上都有显著提高。

参 考 文 献

1 Diffie W, Hellman M E. New directions in cryptography. IEEE Trans Information Theory, 1976, 22: 644~654

2 Diffie W, Hellman M E. Multiuser cryptographic techniques. In: Proc of AFIPS National Computer Conference. 1976, 109~116

3 Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 1978, 21: 120~126

4 ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. Advances in Cryptology Proceedings of CRYPTO'84. 1985, 10~18

5 Simmons G J. Sysmmetric and asymmetric encryption. ACM Computing Surveys, 1979, 11(4): 305~330

6 卢开澄. 计算机密码学. 北京: 清华大学出版社, 1990

7 Schnorr C P. Efficient signature generation by smart cards. Journal of Cryptology, 1991, 4: 161～174
8 Ong H, Schnorr C P. Fast signature generation with a Fiat Shamir-like scheme. Advances in Cryptology-Eurocrypt'90, 1990, 473: 432～440
9 Florent C, Antoine J. Differential collisions in SHA-0. Advances in Cryptology-CRYPTO'98, 1998, 56～71
10 Henri G, Helena H. Security analysis of SHA-256 and sisters. Selected Areas in Cryptography 2003, 2003, 175～193
11 Chaum D. Blind signatures for untraceable payments. Advances in Cryptology-. CRYPT0'82, Plenum Press. 1983, 199～203
12 Chaum D. Untraceable electronic mail, return address and digital pseudonyms. Communications of the ACM, 1981, 124(2): 4～88
13 Juels A, Luby M, Ostrovsky R. Security of blind digital signatures. Advances in Cryptology-CRYPTO'97, Springer-Verlag. 1997, 150～164
14 Abe M, Okamoto T. Provably secure partially blind signatures. Advances in Cryptology-Crypto 2000, Springer-Verlag. 2000, 271～286
15 Maitland G, Boyd C. A provably secure restrictive partially blind signature scheme—Public key cryptography. International Workshop on Practice and Theory in PKC 2002, Springer-Verlag. 2002, 99～114
16 Chaum D, Fiat A, Naor M. Untraceable electronic cash. Advances in Cryptology-Crypto'88, Springer-Verlag. 1988, 319～327
17 Franklin M, Yung M. Secure and efficient off-line digital money. Proc of ICALP'93, Springer-Verlag. 1993, 700: 265～276
18 Camenisch J, Maurer U, Stadler M. Digital payment systems with passive anonymity-revoking trustees. Journal of Computer Security, 1997, 5(1): 69～89
19 Ferguson N. Single term off-line coins. In: Helleseth T, ed. Advances in Cryptology-Eurocypt'93, Springer-Verlag. 1994, 318～328
20 Ferguson N. Extensions of single-term off-line coins. Advances in Cryptology-Crypto'93, Springer-Verlag. 1993, 292～301
21 Yacobi Y. Efficient electronic money. In: Proc of Asiacrypt'94, Springer-Verlag. 1994, 917: 153～163
22 Chaum D, Antwerpen V H. Undeniable signatures. Advances in Cryptology-Crypto'89, Springer-Verlag. 1990, 212～216
23 Chaum D. Zero-knowledge undeniable signatures. Advances in Cryptology Eurocrypt'90, Springer-Verlag. 1991, 458～464
24 Chaum D. Designated confirmer signatures. Advances in Cryptology Eurocrypt'94, Springer-Verlag. 1995, 86～91
25 Okamoto T. Designated confirmer signatures and public key encryption are equivalent. Advances in Cryptology-Crypto'94, Springer-Verlag. 1994, 61～74
26 Michels M, Stadler M. Efficient convertible signature schemes. In: Proc 4th Workshop on Selected Areas in Cryptography(SAC'97), 1997, 231～244
27 Michels M, Stadler M. Generic constructions for secure and efficient confirmer signature schemes. Advances in Cryptology-Crypto'98, Springer-Verlag. 1998, 406～421
28 Camenisch J, Michels M. Confirmer signature schemes secure against adaptive adversaries. Advances in

Cryptology-Crypto'2000, Springer-Verlag. 2000, 243～258
29 Mambo M, Usuda K, Okamoto E. Proxy signature: Delegation of the power to sign messages. IEICE Trans Fundam, 1996, E79-A(9): 1338～1354
30 Mambo M, Usuda K, Okamoto E. Proxy signature for delegating signing operation. In: Proc 3rd ACM Conference on Computer and Communications Security, ACM Press. 1996, 48～57
31 Neuman B C. Proxy-based authorization and accounting for distrubuted systems. In: Proc 13th International Conference on Distrubuted Systems, 1993, 283～291
32 Zhang Fangguo, Kim Kwangjo. Threshold proxy signature schemes. 1977 Information Security Workshop, Japan. 1977, 191～197
33 Desmedt Y. Society and group oriented cryptography: A new concept. Advances in Cryptology-Crypto'87, Springer-Verlag. 1988, 120～127
34 Chaum D, Heyst H V. Group signatures. Advances in Cryptology-Eurocrypt'91, Springer-Verlag. 1991, 257～265
35 Taore J. Group signatures and their relence to privacy-protecting off-line electronic cash systems. In: Australasian Conference on Information Security and Privacy (ACISP'99), LNCS 1587, Springer-Verlag. 1999, 228～243
36 Jeong I R, Lee D H, Lim J I. Efficient transferable cash with group signature. In: Davida G I, Frankel Y, eds. Information Security, LNCS 2200, Springer-Verlag. 2001, 462～474
37 Hwang T. Cryptosystem for group oriented cryptography. In: Advances in Cryptology Proceedings of Eurocrypt'90. 1990, 352～360
38 Miller V. Use of elliptic curves in cryptography. Advances in Cryptology-Crypto'85, Springer-Verlag. 1986, 417～426
39 Koblitz N. Elliptic curve cryptosystems. Mathematics of Computation, 1987, 48(177): 203～209
40 Koblitz N. Algebraic Aspects of Cryptography. Springer-Verlag (ACM 3). 1998
41 Koblitz N, Menezes A, Vanstone S. The state of elliptic curve cryptography. Designs, Codes and Cryptography. 2000, 19: 173～193
42 Silberman J H. The arithmetic of elliptic curves. Springer-Verlag. 1986, 46～61, 130～136
43 Raju G V S; Akbani R. Elliptic curve cryptosystem and its applications. IEEE International Conference on Systems, Man and Cybernetics, 2003, 2(5): 1540～1543
44 Schoof R. Elliptic curve over finite fields and computation of square roots mod p. Mathematics of Computation, 1985, 44: 483～494
45 Elkies N D. Elliptic and modular curves over finite fields and related computational issues. In: Buell D A, Teitelbaum J T, eds. Computational Perspective on Number Theory. AMS/International Press. 1998, 21～76
46 Atkin A O. The number of points on an elliptic curve modulo a prime (ii). Draft. 1992
47 Atkin A O, Morain F. Finding suitable curves for the elliptic curve method of factorization. Journal of Math Comp, 1993, 60: 399～405
48 Shanks D. Five number theoretical algorithms. In: Proc 2nd Manitoba Conference on Numerical Math (Congresses Numerantium Ⅶ, Univ. Manltoba Winnipeg). 1972, 353～356
49 Terr D. A modification of Shanks' baby-step giant-step algorithm. Mathematics of Computation. 1999, 767～773
50 Pollard J M. Monte Carlo methods for index computation (mod p). Mathematics of Computation, 1978, 32: 918～924

51　Escoot A, Sager J, Selkirka, et al. Attacking elliptic curve: Cryptosystems using the parallel Pollard RHO method. CryptoBytes—The Technical News Letter of RSA Laboratories. 1999, 15～19
52　Oorschot V P, Wiener M. Parallel collision search with cryptanalytic applications. Journal of Cryptology, 1994, 12(1): 1～28
53　Pohlig S C, Hellman M E. An improve algorithm for computing logarithms over GF(p) and its cryptographic significance. IEEE Trans on Information Theory, 1978, 24: 106～110
54　Menezes A, Okamoto T, Vanstone S. Reducing elliptic curve logarithms to logarithms in a finite field. IEEE Trans on Information Theory, 1993, 39: 1639～1646
55　Balasubramanian R, Koblitz N. The improbability that an elliptic curve has subexponential discrete log problem under the Menezes-Okamoto-Vanstone algorithm. Journal of Cryptology, 1998, 11(2): 141～145
56　Saito T, Uchiyama S. A remark on the MOV algorithm for non-supersingular elliptic curves. IEICE Trans Fundamentals, 2001, E84-A(5): 1266～1268
57　Harasaea R, Shikata J, Suzuki J, et al. Comparing the MOV and FR reductions in elliptic curve cryptography. In: Proc of Eurocrypto'99, Springer-Verlag. 1999, 1592: 190～205
58　Semaev I A. Evaluation of discrete logarithms on some elliptic curves. Math Comp, 1998, 67: 353～356
59　Satoh T, Araki K. Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves. Comm Math Univ Sancti Pauli, 1998, 47: 81～92
60　IEEE P1363: Standard for Public Key Cryptography. Working Draft, 1998
61　http://rfc.net/rfc3278.html
62　Lehmann F J, Maurer M, Muller V, et al. Counting the number of points on elliptic curves over finite fields of characteristic greater than three. In: Proceedings of Algorithmic Number Theory Symposium I, Springer-Verlag, 1994, 60～70
63　Couveignes J M, Morain F. Schoof's algorithm and isogeny cycles. 1st Algorithmic Number Theory Symposium-Cornell University, Springer-Verlag, 1994, 43～58
64　Menezes A, Vanstone S, Zuccherato R. Counting points on elliptic curves over $F(2^m)$. Mathematics of Computation, 1993, 60: 407～420
65　Lercier R, Morain F. Computing isogenies between elliptic curves over Fq using Couveignes's algorithm. Journal of Math Comp, 2000, 69: 351～370
66　Lercier R, Morain F. Counting the number of points on elliptic curves over finite fields: Strategies and performances. In: Proc of Cryptology-Crypto'92, Springer-Verlag. 1993, 79～94
67　Satoh T, Skjernaa B, Taguchi Y. Fast computation of canonical lifts of elliptic curves and its application to point counting. Finite Fields and Their Applicatons, 2003, 9(1): 89～101
68　Gaudry P. A comparison and a combination of SST and AGM algorithms for counting points of elliptic curves in characteristic 2. In: Proceedings of Asiacrypt 2002, Springer-Verlag. 2002, 2501: 311～327
69　Satoh T. On p-adic point counting algorithms for elliptic curves over finite fields. In: Algorithmic Number Theory, 5th International Symposium, ANTS-V, Springer-Verlag, 2002, 7: 43～66
70　http://research.sun.com/projects/crypto/performance.pdf
71　Lauter K. The advantages of elliptic curve cryptography for wireless security. IEEE Wireless Communications, 2004, 11(1): 62～67
72　http://www.drgsf.com/ABmComm3-01.pdf
73　张险峰，秦志光，刘锦德. 椭圆曲线加密系统的性能分析. 电子科技大学学报，2001，30(2)：144～147
74　Harn L. Group-oriented (t, n) threshold digital signature scheme and multisignature. IEEE

Proceedings on Computers and Digital Techniques，1994，141(5)：307～313
75 Chen L，Pedersen T P. New group signature schemes. Advances in Cryptology-Eurocrypt ' 94，Springer-Verlag. 1995，171～181
76 Lee W B，Chang C C. Efficient group signature scheme based on the discrete logarithm. IEE Proceeding on Computers and Digital Techniques，1998，145(1)：15～18
77 Tseng Y M，Jan J K. Improved group signature scheme based on the discrete logarithm problem. Electronics Letters，1999，35(1)：37～38
78 Camenisch J. Group signature schemes and payment systems based on the discrete logarithm problem [Ph D dissertation]. Hartung Gorre Verlag Konstanz. 1998
79 Ateniese G，Camenlsch J，Joye M，et al. A practical and provably secure coalition-resistant group signature scheme. Advances in Cryptology-Crypto 2000. 2000，1880：255～270
80 Desmedt Y，Frankel Y. Threshold cryptosystems. Advances in Cryptology-Crypto ' 89，Springer-Verlag. 1990，307～315
81 Desmedt Y，Frankel Y. Share generation of authenticators and signature. Advances in Cryptology-Crypto'91，Springer-Verlag. 1992，457～469
82 Adballa M，Miner S，Namprempre C. Forward-secure threshold signature schemes. Topics in Cryptology-CT-RSA 2001，Springer-Verlag. 2001，441～456
83 Li C M.，Hwang T，Lee N Y. (t, n) threshold signature schemes based on discrete logarithm. Advances in Cryptology-Eurocrypt'94，Springer-Verlag，1995，191～200
84 Shamir A. How to share a secret. Communications of the ACM，1979，24(11)：612～613
85 Blakley G R. Safeguarding cryptographic keys. In：Proc of the National Computer Conference，American Federation of Information Processing Societies. 1978，48：313～317
86 Harn L，Yang S. Group-oriented undeniable signature schemes without the assistance of a mutually trusted party. In：Advances in Cryptology Proceedings of Auscrypt，Springer-Verlag. 1993，133～142
87 Pedersen T P. A threshold cryptosystem without a trusted party. In：Advances in Cryptology Proceedings of Eurocrypt. 1991，522～526
88 Ingemarsson I，Simmons G L. A protocol to set up shared secret schemes without the assistance of a mutually trusted party. In：Advances in Cryptology Proceedings of Eurocrypt，1990，266～282
89 Frankel Y. A practical protocol for large group oriented networks. In：Advances in Cryptology Proceedings of Eurocrypt，1989，56～61
90 Chang Y S，Wu T C，Huang S C. ElGamal-like digital signature and multisignature schemes using self-certified public keys. Journal of Systems and Software，2000，50(2)：99～105
91 Wu T S，Hsu C L. Threshold signature scheme using self-certified public keys. Journal of Systems and Software，2003，67(2)：89～97
92 Shamir A. How to share a secret. Communications of the ACM，1979，22(11)：612～613
93 The digital signature standard proposed by NIST. Commmunications of the ACM，1992，35(7)：36～40
94 Chen T S，Huang K H，Chung Y F. A division-of-labor-signature (t, n) threshold authenticated encryption scheme with message linkage based on the elliptic curve cryptosystem. In：Proc IEEE International Conference on e-Technology，e-Commerce and e-Service. 2004，106～112
95 Koblitz N，Menezes A，Vanstone S. The State of elliptic curve cryptography. Designs，Codes and Cryptography，2000，19：173～193
96 Knuth D F. The Art of Computer Programming，Volume 2：Seminumerical Algorithms. 2nd edn. Reading，MA：Addison-Wesley，1981，257～326

Chapter

第 4 章

密钥管理

安全的组通信具有广泛的应用领域,而组密钥管理算法一直是安全组通信研究中的热点问题。本章针对不可靠、开放的组通信环境下密钥分发管理机制这一热点问题进行了探讨,重点对组密钥分发协议的鲁棒性、可扩展性和动态性这三方面的特性进行了深入而广泛的研究。首先概述了当前安全组通信方案的研究现状,然后基于信息熵的基本概念,形式化地定义和描述了 B-GKDS,S-GKDS 和 S-GKDS-TL 这 3 种组密钥分发协议的模型,安全和性能分析证明,协议能够有效地保证组密钥的前向/后向隐私性,抗同谋破解和组机密性。结合具体应用网络环境的特点,如无线网络、移动网络(NEMO 网)和无线传感器网络,分别探讨了 S-GKDS 协议和 S-GKDS-TL 协议在这些具体网络环境中的应用问题,着重分析协议在这些具体网络中的安全性、可扩展性和动态稳定性。最后,对无线传感器网络中典型的密钥管理方案进行研究,给出这些方案的综合分析和所需解决的问题。

4.1 研究背景

在数据网络传输过程中,可以通过加密/解密机制对数据进行保护,以满足传输过程中数据的私有性(privacy)、机密性(confidentiality)、完整性(integrity)的要求。数据发送方和接收方掌握相应的加密/解密密钥。发送方在发送数据之前,首先用加密密钥对原始数据(明文)进行加密,然后将密文通过网络发送给接收方。接收方接收到密文后,利用解密密钥进行相应的解密操作,还原出原始数据,完成数据的安全传输。安全通信中一般采用对称加密体制。由于受到加密运算强度的限制,公钥体制一般只用于对称密钥的分发。

密钥是加密/解密机制中的基本条件,是实现安全数据传输的基础。如何有效地实现密钥的创建(协商)、发布、安装、定期更新、事件驱动更新(在组通信系统中,组成员关系发生变化,如用户加入、退出等,所引发的密钥更新),避免泄露和被窃听,是保证数据安全的关键性问题,即密钥管理机制问题。通过对密钥管理机制的研究,可以有效、安全、可靠地为安全数据通信中的加密/解密操作提供密钥支持。密钥管理机制的优劣直接关系到安全数据传输的成败、稳定性和可靠性。所以研究密钥管理机制,寻求高效、安全、可靠的解决方案是至关重要的。

在安全单播情况下,会话密钥(session key,也称为数据加密密钥)的分发管理方式一般

有：手工分发、KDC 集中分发、基于公钥体制分发和分布式密钥协商等；除安全单播外，一对多、多对多安全组通信机制具有广泛的应用领域，如付费视频点播、安全视频会议、网络协作等。通过高效、安全的密钥管理方案的实施，对组密钥的生成、更新进行相应管理，可以实现对组通信数据的访问控制，保证只有掌握合法组密钥的用户才能对数据进行访问，从而保证数据的机密性、私有性等安全特性。安全组通信密钥管理以安全单播密钥分发管理为基础。

安全的组密钥管理方案主要分为 3 类：集中式、分散式和分布式。在集中式密钥管理方案中，由单一通信实体担当中央控制节点，全权负责组密钥的创建、分发和组成员关系发生变化时的密钥更新。在分散式组通信密钥管理中，整个通信组分成若干子组，每个子组分别由不同的子组控制器管理。所有子组之间可以是分布关系，或是由一个中央控制节点在最高层进行集中控制。而在分布式的密钥管理方案中，没有中央控制节，各节点都是独立的通信实体，他们共同参与通信组的安全认证和组密钥的创建及更新。这种管理方式很容易推广到 peer-to-peer 应用。

就组通信的应用背景问题来看，当前，Internet 已经成为现代社会赖以存在和发展的重要基础设施。它迅速进入了国民经济、社会生活的方方面面，发挥着不可替代的作用，对人类的生活方式、工作方式和思维方式都产生着深远的影响。Internet 在推动全球社会进入信息化时代的同时，其自身也在不断地演变以满足各种日新月异的需求。近几年来，Internet 的带宽增长迅速，覆盖范围几乎遍及全球。这些进步使得很多业务的开展成为可能。

而且，随着移动通信和普适计算的兴起，计算模式正朝着“深入生活、无处不在”的方向发展。无线传感器网络（wireless sensor networks，WSN）和移动网络（network mobility，NEMO）正是当前两种重要的普适计算环境。

作为传统骨干网络边界的延伸，无线传感器网络是集成传感器、微机电系统和网络三大技术而形成的一种全新的信息获取和处理技术。当前，以廉价而低功耗计算设备代表的“后 PC 时代”冲破了传统台式计算机和高性能服务器的设计模式；网络的普遍化带来的计算处理能力是难以估量的；微机电系统（micro-electro-mechanism system，MEMS）的迅速发展奠定了设计和实现片上系统（system on chip，SOC）的基础，并使得低成本、低功耗、微体积传感器节点得以广泛使用。这种微传感器节点由传感单元、数据处理单元、通信单元和便携式电源组成[1]，能够完成数据采集、信号监测和传送信息的任务。传感器网络即为以上 3 方面高度集成而孕育出新的信息获取和处理模式。

无线传感器网络（WSN）则是由一组传感器节点通过无线介质连接而构成的无线网络，它是随着传感器技术和通信技术的发展而出现的，并因为其应用的广泛性而越来越多地得到业界的高度重视。WSN 采用自组织方式配置大量微型的智能传感节点，通过节点的协同工作来采集和处理网络覆盖区域中的目标信息。无线传感器网络具有许多得天独厚的技术优势，它在环境与军事监控，地震与气候预测，地下、深水以及外层空间探索等许多方面已展示出广泛的应用前景。可以说，无线传感器网络是信息感知和采集的一场革命，是 21 世纪最为重要的技术之一。

另外，相对于传统的无线移动通信，下一代无线网络应提供给用户更高的宽带服务，并且透明地将技术集成到系统环境中，从而实现位置无关性。这样就需要整合异构网络和协

议。移动网络 NEMO 正是这种异构体系结构中不可或缺的一部分。IETF 的 NEMO 工作组认为,移动网是一个具有 Internet 接入点的独立单元,也可以认为它是一个叶子网络。NEMO 不仅要求提供和主干网络连接的功能,而且还需要具有用户位置注册和用户位置发现的功能。因此,下一代无线网络所具备的另一个特征是多地址和在多域环境中的移动性。这就要求下一代网络为用户提供全球范围内的无缝连接,使得用户可以在任何时候、任何地点使用最适合的接口接入网络。

传统的移动 IP 所提供的漫游机制,仅局限于主机(host)漫游的无线移动网络。当主机扩展成网络时,即为具有移动网络 NEMO 特性的网络。例如,在公交设施上设置一台移动路由器(mobile router),此路由器通过上行接口对外连接互联网,对内通过无线局域网向移动用户提供接入服务。当公共交通设施移动漫游时,其中的所有移动用户将被视为一个移动子网。此时,移动用户设备并不需要单独执行漫游和无线切分(handoff)服务,取而代之的是通过移动路由器执行漫游与切分服务来保持网络整体的畅通。

所有的这些实际应用背景,无论是 Internet,还是无线传感器网络和移动网络,都对安全组通信协议的设计带来了颇多挑战,如开放设计的 Internet、易受入侵和非法攻击的移动网络、不可靠的无线传感器网络等因素。因此,以 Internet 为核心、移动计算和无线通信为边缘延伸的普适计算环境将带来一些新的计算模式。这使得组通信的应用将变得更加普遍,同时也对安全的组通信协议设计提出了许多新的亟待解决的课题。

考虑到这些新的网络形态和计算模式的出现,以及组通信技术在未来移动通信和普适计算环境中更为广泛的应用前景,我们有必要结合具体的计算背景环境对安全组通信的组密钥管理机制进行深入研究。

从 1998 年 IETF 成立组播安全研究小组以来,组通信的安全和性能问题一直是一个重要的热点问题。该主题历年来一直为通信和网络领域一些重要的国际学术会议所关注,如 ACM SIGCOMM,ACM CCS,IEEE INFOCOM,IEEE GlobeCom 和 IEEE ISSP(IEEE Symposium on Security and Privacy)等。尽管该主题目前已取得了很多研究成果,然而,目前组通信密钥管理研究仍然存在许多亟待解决的问题。

首先,现有的组密钥管理方案多限于协议本身的安全性和性能方面的研究,存在着与具体应用结合研究不够等问题。如何把组密钥管理的研究与具体应用相结合,如安全的视频点播、P2P 应用、无线传感器网络、移动网络应用等,以寻求最佳的密钥管理方案,仍具有重要的研究意义和实用价值。

其次,就组密钥管理方案本身而言,根据当前组密钥管理研究的热点问题和研究趋势来看,对如下 3 个主题展开深入的研究仍很有必要性:

(1) 组密钥分发协议的鲁棒性:如何在不可靠、易受攻击和开放的组通信环境中保证组密钥能被可靠地分发和更新,也即要求协议具备一定的容侵、容错特性。

(2) 组密钥分发协议的可扩展性:如何对较大规模的动态组用户群进行有效的组密钥管理。

(3) 组密钥分发协议的动态性能:如何在组用户频繁参与或退出组通信的情形下保持协议的动态稳定性。

围绕如上 3 个主题的研究,最近几年已有很大进展。但考虑到组通信的实际复杂性,组

密钥管理的研究依然存在很大困难，并有很大的改善空间。组密钥分发协议的鲁棒性、可扩展性和动态性能仍然是组密钥管理研究工作中亟待解决的几个重要问题，特别是在复杂、开放、不可靠和易受攻击的普适计算环境中。因此，本章的主要工作是进一步深入开展这方面的研究。图 4.1.1 描述了本章的研究层次。我们采用的研究策略是循序渐进的，最终构建一个具有良好鲁棒性、可扩展性和动态性能的组密钥分发协议。

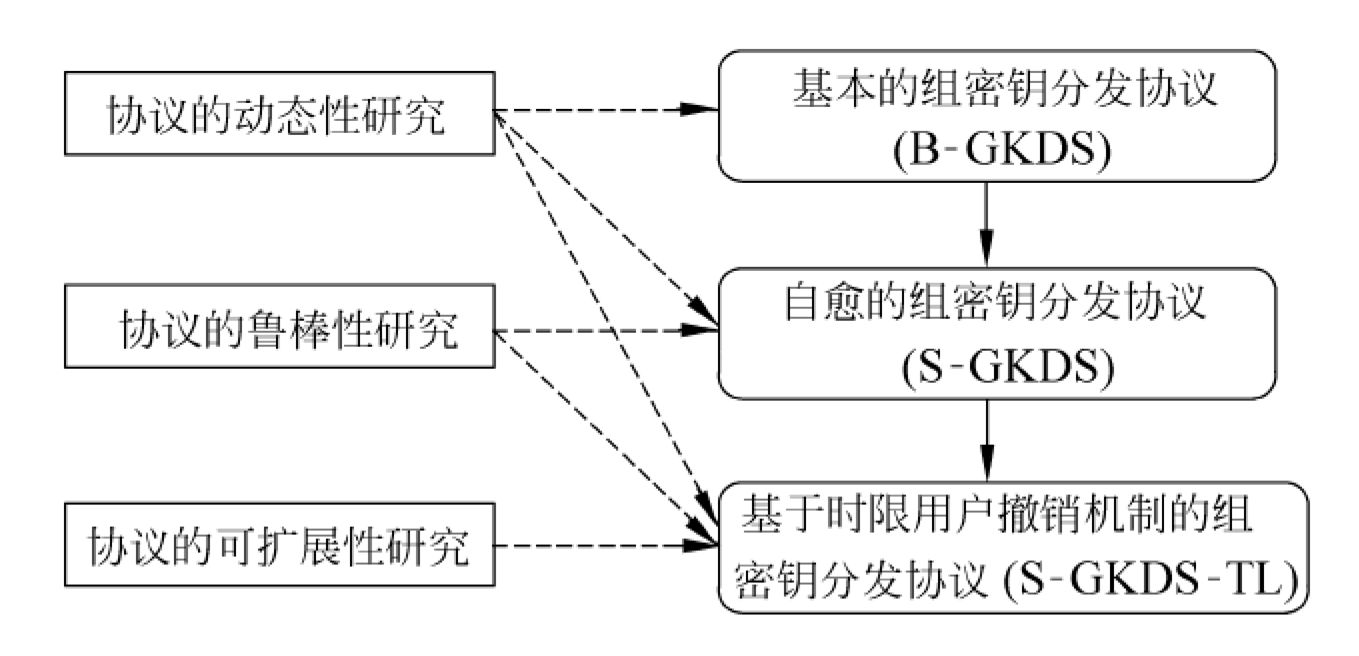


图 4.1.1　组密钥分发协议的研究层次

组密钥分发协议的研究难点是在组通信密钥管理中，经常需要在协议的管理效率和计算开销、安全性、可靠性（密钥恢复）、存储开销和网络带宽占用等诸多因素之间进行综合权衡[2～5]。通过对密钥管理协议特性之间的权衡研究可以使密钥管理方案在诸多方面之间进行取舍，最终找到一种满足特定应用需求的实际可行方案。

4.2　组密钥分发机制研究综述

4.2.1　概述

组通信被广泛应用于多种网络业务中，能够有效地实现一对多、多对多的信息交换。安全的组通信通过引入对称加解密技术实现对组通信数据的访问控制，使得只有掌握合法组密钥的用户才能对组通信数据进行访问，从而保证数据的机密性（confidentiality）和私有性（privacy）。安全组通信可以有效地应用于数据安全敏感的网络业务中，提供高效、安全的数据服务，如信息软件分发、安全多媒体数据传输、付费视频点播（pay-per-view）和安全视频会议等。

在安全组通信系统中，密钥管理充当着重要角色。各组内用户通过高效、安全的密钥管理机制的实施，可以对密钥生成、密钥分发、密钥协商及密钥更新等进行有效的管理，为安全的组通信进行加密、解密操作提供基本的安全保障。密钥管理机制的优劣直接关系到数据传输的安全性（前向保密、后向保密、抗同谋破解等的安全特性）、稳定性和可靠性。所以，密钥管理机制是安全组通信中至关重要的问题，对其研究具有重要的意义。

目前，安全组通信密钥管理方案可以分为集中式管理、分散式管理和分布式管理 3 类[6,7]：①集中式组通信密钥管理：在集中式组通信密钥管理中，由单一通信实体担当中央控制节点全权负责通信组密钥的创建、分发和组成员关系发生变化时的密钥更新。集中式组通信密钥管理方案可以分为平面管理模式和层次树管理模式两种类型。层次树管理方案

采用密钥逻辑树对密钥进行管理，提高了密钥管理方案的可扩展性和高效性。②分散式组通信密钥管理：在分散式组通信密钥管理中，整个通信组分成若干子组，每个子组分别由不同的子组控制器管理。所有子组之间可以是分布关系，或是由一个中央控制节点在最高层进行集中控制。根据通信组是否拥有唯一的组密钥，分散式组通信密钥管理可以分为密钥分发服务器模式和重加密服务器模式两种方式。③分布式组通信密钥管理：在分布式组通信密钥管理中，没有中央控制节点和子组控制节点，每个节点都是一个独立的通信实体，它们共同参与通信组的安全认证和通信组密钥的创建。根据是否所有用户参与组密钥协商，分布式组密钥管理方案可以分为非协商模式和协商模式两类。

在组通信密钥的管理过程中，可以根据不同应用需求采用不同的密钥管理方案来达到相应的管理目标。如密钥可以在密钥分发中心（key distribution center，KDC）生成，然后进行组内分发；也可以基于 DH 算法的支持，通过组内成员共同协商确定组通信密钥；或者在通信组成员关系发生变化时，通过单向函数在本地计算出新密钥。应针对具体应用特点，确定管理方案以满足系统不同的安全性需求：如付费视频点播系统，除对付费用户进行身份认证，需同时确保无冒名顶替等问题，而无需对数据源进行认证；对于安全视频会议系统，除对数据接收方进行身份认证以外，同时还要对数据发送方（组通信源）进行认证，确保无抵赖。由此可见，不同的安全组通信系统具有不同的特性要求。结合网络应用特点，有针对性地进行组通信密钥管理机制的研究，以寻求高效、安全、可靠的安全组通信密钥管理的解决方案，具有重要的研究价值和实际意义。

4.2.2 组密钥管理方案的特性需求

鉴于安全组通信系统中数据传输本身所具有的固有特点和要求，密钥管理机制应满足高效性、可扩展性、可靠性、鲁棒性、安全性（前向保密、后向保密、抗同谋破解、身份验证等）等特性要求。

4.2.2.1 高效性和可扩展性

高效性（efficiency）与可扩展性（scalability）密切相关：高效性是指以最小的系统资源开销（包括存储开销、计算开销、网络带宽开销）实现组通信密钥管理；而可扩展性是指组通信中采用的密钥管理方案是否适用于大规模通信组，即当通信组成员数量增加时，密钥管理方案能否胜任。一般情况下，高效的密钥管理方案其可扩展性也好。

在安全组通信系统中，应尽量减小密钥管理所带来的网络带宽开销，提高网络传输效率，节约出带宽供数据传输使用。此外，还应该尽量减小密钥在各节点保存所需的存储开销、加密/解密操作及辅助运算产生的计算开销。衡量密钥管理方案高效性的常用相关性能参数如下：

（1）KDC 和各用户节点所需保存的密钥数；

（2）用户离开时，密钥更新的消息长度；

（3）用户加入时，密钥更新长度（一般分为组播的消息长度和单播的消息长度）；

（4）密钥更新时，所需的加密/解密运算次数；

（5）密钥更新时，所需的辅助运算的强度和次数（如单向函数运算、混合函数等）；

（6）密钥协商方式中，密钥协商所需的交互的轮次数、幂运算的次数。

4.2.2.2　安全性

在安全组通信系统中，由于用户成员关系经常处于动态变化中，如不断有新用户加入通信组、老用户退出通信组等，所以组通信密钥管理在安全性需求方面具有不同于单播密钥管理的特点。在安全组通信系统中，密钥管理方案应满足前向隐私性、后向隐私性、抗同谋破解、用户身份认证等安全特性要求。

前向隐私性：确保离开通信组的用户节点不能继续参与安全通信组的数据传输，即无法利用其所掌握的密钥解密后继组通信数据和生成有效的加密报文。离开的节点包括主动退出通信组的节点和被强制退出的恶意节点。保证前向隐私性的方法是在有用户离开通信组时，及时更新通信组密钥。

后向隐私性：确保新加入的组成员无法利用现有密钥破解其加入通信组前的组通信数据。保证前向隐私性的方法是在有新用户加入通信组时，及时更新通信组密钥。

抗同谋破解：安全组通信密钥管理不仅要防止某个节点破解系统，还要防止某几个节点联合起来破解。如果几个恶意节点联合起来，掌握了足够多的密钥信息，则对密钥管理系统造成危害或破坏密钥管理机制，使得无论系统如何更新密钥它都可以获得更新的密钥，导致组通信中密钥管理的前向保密和后向保密失败，或者使得恶意节点可以冒充其他节点进行欺骗（破解系统的认证功能），我们把这种情况称为同谋破解。安全组通信密钥管理系统应杜绝同谋破解或尽量降低同谋破解的概率。

用户身份认证[8]：在安全组通信中，数据源和通信组成员身份的认证是重要的安全特性，包括用户身份认证和数据源认证。通过通信组成员的身份验证，确保只有具有合法身份的用户才可以加入通信组，参与通信组内的数据传输，确保非组成员无法生成有效的认证信息，进而无法冒充组成员接收组通信报文，可以拒非法企图窃取组通信数据者于通信组之外。通过数据源身份认证可以确保其身份的合法性，确保数据源对发送的数据负责，保证不可否认性。

在安全组通信网络应用中，不同的应用对于安全性的要求具有差异性，如成员关系相对较为固定的安全视频会议系统对于前向、后向保密安全性要求较低，而对于数据源和用户身份的认证要求相对较高；但在付费视频点播应用中，对前向、后向保密安全性以及用户身份认证要求较高，而对数据源认证要求相对较低。所以，我们应根据应用需求保证安全性。

4.2.2.3　可靠性

可靠性是指密钥管理系统能否在不可靠的组通信中保证密钥更新消息的可靠传输，能否在密钥更新消息丢失、节点发生未可预期错误的情况下恢复正常的安全组数据传输。

如何避免各种管理方案中的缺陷，弥补不足，提高组通信系统的可靠性是至关重要的。如在安全组通信密钥管理的过程中引入延时重发和消息握手机制，在密钥分发、更新等过程中确保消息可靠传递，或引入密钥恢复机制提高密钥更新的可靠性都是值得研究的问题。此外，在密钥管理方案中引入密钥恢复机制[9~13]，保证密钥更新的可靠性。在多媒体安全组通信中，密钥管理方案的可靠性与密钥的传送方式直接相关。密钥传送方式可以分为两种[14]：媒体无关信道和媒体相关信道方式。采用媒体无关信道，由安全系统本身通过可靠

组播或者密钥恢复机制保证密钥管理的可靠性;采用媒体相关信道,必须在应用层采取相应的可靠性保证机制,保证密钥管理的可靠性。

4.2.2.4 鲁棒性

对于单播来说,通信的任何一方失败都会使会话终止,而组通信中部分节点的失败不应影响整个组通信会话的继续进行,这就对组通信密钥管理提出了鲁棒性的要求。鲁棒性问题是指单一节点或者少数节点失效是否影响整个系统的运作,是否存在 1-affect-n 问题[15],如集中式密钥管理方案中的密钥分发中心 KDC 单失效节点问题。IOLUS[16] 等密钥管理方案采用分散式管理,较好地解决了 1-affect-n 问题,提高了系统的鲁棒性。

在安全组通信密钥管理中,要根据实际应用对上述特性的不同要求,在各特性之间进行权衡和折中,确定适合该应用中的最佳密钥管理方案。

4.2.3 组密钥管理方案分类

目前,安全组通信密钥管理方案可以分为集中式管理、分散式管理和分布式管理 3 类。

1. 集中式组通信密钥管理

由单一通信实体担当中央控制节点(又称为组控制器(group controller,GC),密钥分发中心 KDC),全权负责通信组密钥的创建、分发和组成员关系发生变化时的密钥更新。集中式组通信密钥管理方案可以分为平面管理模式(centralized flat)和层次树管理模式(hierarchy tree)两种类型。层次树管理方案采用密钥逻辑树对密钥进行管理,提高了密钥管理方案的可扩展性和高效性。在基本 LKH(logical key hierarchy)[15,17] 管理方案的基础上,通过对诸多特性间进行权衡和改进(如通过减小密钥更新消息长度以提高网络传输效率、提高密钥协商机制的可靠性、增加密钥信息自恢复能力等)可得到许多变种管理方案。

2. 分散式组通信密钥管理

在分散式组通信密钥管理中,整个通信组分成若干子组,每个子组分别由不同的子组控制器管理。所有子组之间可以是分布关系,或是由一个中央控制节点在最高层进行集中控制。根据通信组是否拥有唯一的组密钥 GK,分散式组通信密钥管理可以分为两种方式,密钥分发服务器模式(key distributed servers scheme)和重加密服务器模式(re-encryption servers scheme),如图 4.2.1 所示。在密钥分发服务器模式中,具有唯一的组密钥 GK,而在重加密服务器模式中,不具有唯一的组密钥 GK。

3. 分布式组通信密钥管理

在分布式组通信密钥管理中,没有中央控制节点和子组控制节点,每个节点都是一个独立的通信实体,它们共同参与通信组的安全认证和组密钥 GK 的创建。这种控制方式很容易推广到 peer-to-peer 的应用模式。根据是否所有用户参与组密钥协商,分布式密钥管理可以分为两类:非协商模式(non-negotiation schemes)和协商模式(negotiation schemes),如图 4.2.1 所示。非协商模式更注重管理的高效性,但它不能保证所有组成员平等地参与组密钥的协商生成。协商模式注重于保证所有的组用户平等地参与组密钥 GK 的协商生成。非协商模式建立在对 LKH[15,17],C-OFT[18],Flat Table[19,20] 等方案的扩展之上,而协商

模式主要是建立在 Diffie-Hellman 密钥协商协议的扩展之上。

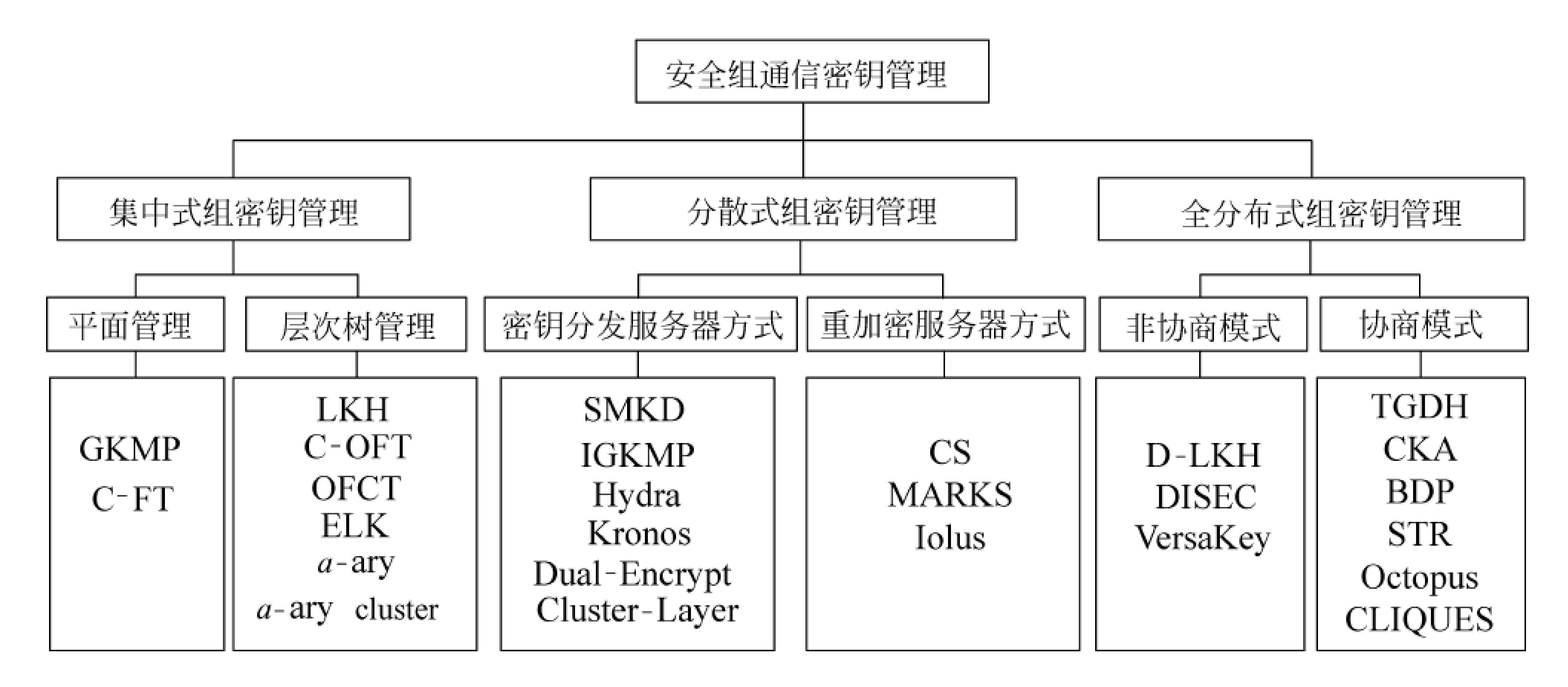


图 4.2.1　安全组通信密钥管理方案分类

4.2.4　集中式组密钥管理

4.2.4.1　平面管理模式

1. 组密钥管理协议

在不需要保证前向隐私性的高效安全组通信中，优先考虑组密钥管理协议(group key management protocol，GCP)[21,22]。GCP 中采用一个密钥分发中心(KDC)集中管理密钥的分发和更新。在通信组建立阶段，KDC 首先在第一个组成员的协助下生成 GKP(group key packet)。GKP 中包含 GTEK(group traffic encryption key)和 GKEK(group key encryption key)两个密钥，GTEK 用于数据加、解密，GKEK 用于加密 GKP 进行密钥分发。然后 KDC 将 GKP 分发给通信组的其他成员，通信组建立完成。

密钥更新：当有新用户加入时，GC 生成一个新 GKP 并用当前 GKEK 加密：(GKP)GKEK，并组播给原通信组成员，同时通过安全单播将新 GKP 发送给新用户，随后通信组使用新的 GTEK 和 GKEK 进行组通信。

安全性：通过密钥更新可以保证后向保密安全性，但除非重建整个通信组，否则无法保证前向保密安全性。因为用户离开后，仍然能够通过先前掌握的密钥信息，继续对通信组中传输的数据进行访问。

2. 集中式平面表管理

在不需要保证抗同谋破解的安全组通信中，可以考虑采用集中式平面表管理(centralized flat table，C-FT)[19,20]。在 C-FT 方案中，使用平面表对密钥进行组织和管理。KDC 中保存一个逻辑密钥表，其中包括 1 个 TEK(traffic encryption key)和 $2w$ 个 KEK(key encryption key)，其中 w 是用户 ID 号的 bit 位数。每个用户节点应保存 TEK(traffic encryption key)和 w 个 KEK，以保证密钥的分发和管理。根据用户节点 ID 编号确定其应保存的密钥集合：

$$\{\text{TEK}\} \cup \{\text{KEK}_{i,j} \mid i\ \text{表示用户编号 bit 位},j\ \text{为该 bit 位的值},j=0,1\}。$$

以 4 位编号为例：支持最大组成员个数为 $2^4=16$，ID 编号由 0000～1111。成员 0101 保存如图 4.2.2 所示密钥。

	TEK	
ID Bit0	$KEK_{0,0}$	$KEK_{0,1}$
ID Bit1	$KEK_{1,0}$	$KEK_{1,1}$
ID Bit2	$KEK_{2,0}$	$KEK_{2,1}$
ID Bit3	$KEK_{3,0}$	$KEK_{3,1}$

(a) KDC 保存的密钥表

	TEK
ID Bit0=0	$KEK_{0,0}$
ID Bit1=1	$KEK_{1,1}$
ID Bit2=0	$KEK_{2,0}$
ID Bit3=1	$KEK_{3,1}$

(b) 成员 0101 保存的密钥表

图 4.2.2　C-FT 方案实例(4 位编号)

为保证前向隐私性，当用户节点离开时，该节点掌握的密钥需要全部更新：更新 TEK 为 TEK′，更新该节点掌握的所有 KEK 为 KEK′。密钥更新消息内容由两部分组成：Part 1 和 Part 2。用所有无需更新的 KEK 加密 TEK′，构成 Part 1；用 TEK′加密所有需要更新的 KEK 对应的新密钥 KEK′，构成 Part 2；组装成密钥更新消息进行组通信发送，完成更新。当用户 U_{0101} 离开时，密钥更新消息内容组成为：

Part 1：$[(TEK')KEK_{0,1}][(TEK')KEK_{1,0}][(TEK')KEK_{2,1}][(TEK')KEK_{3,0}]$

Part 2：$[(KEK'_{0,0})TEK'][(KEK'_{1,1})TEK'][(KEK'_{2,0})TEK'][(KEK'_{3,1})TEK']$

安全性：C-FT 方案可以通过密钥更新保证前向和后向隐私性，但不具有抗同谋破解能力：如用户 U_{0101} 和 U_{1010} 同谋，可以获知所有密钥，对密钥管理系统构成危害。

4242　层次树管理模式

1. 逻辑密钥层次树

逻辑密钥层次树管理方案(logical key hierarchy，LKH)[15,17]具有良好的可扩展性，并可确保前向加密、后向加密、抗同谋破解等安全性。在 LKH 方案中，采用存储在 KDC 中的逻辑密钥树对组通信密钥进行管理。其中使用两种密钥：组密钥 GK(group key)，又称为会话密钥 SK(session key)或数据加密密钥 DEK(data encryption key)，用来加密组通信数据；密钥加密密钥用来进行组密钥更新的辅助密钥。

图 4.2.3 表示一棵二叉平衡 LKH 树。圆形节点均为逻辑密钥节点，方形节点为实际

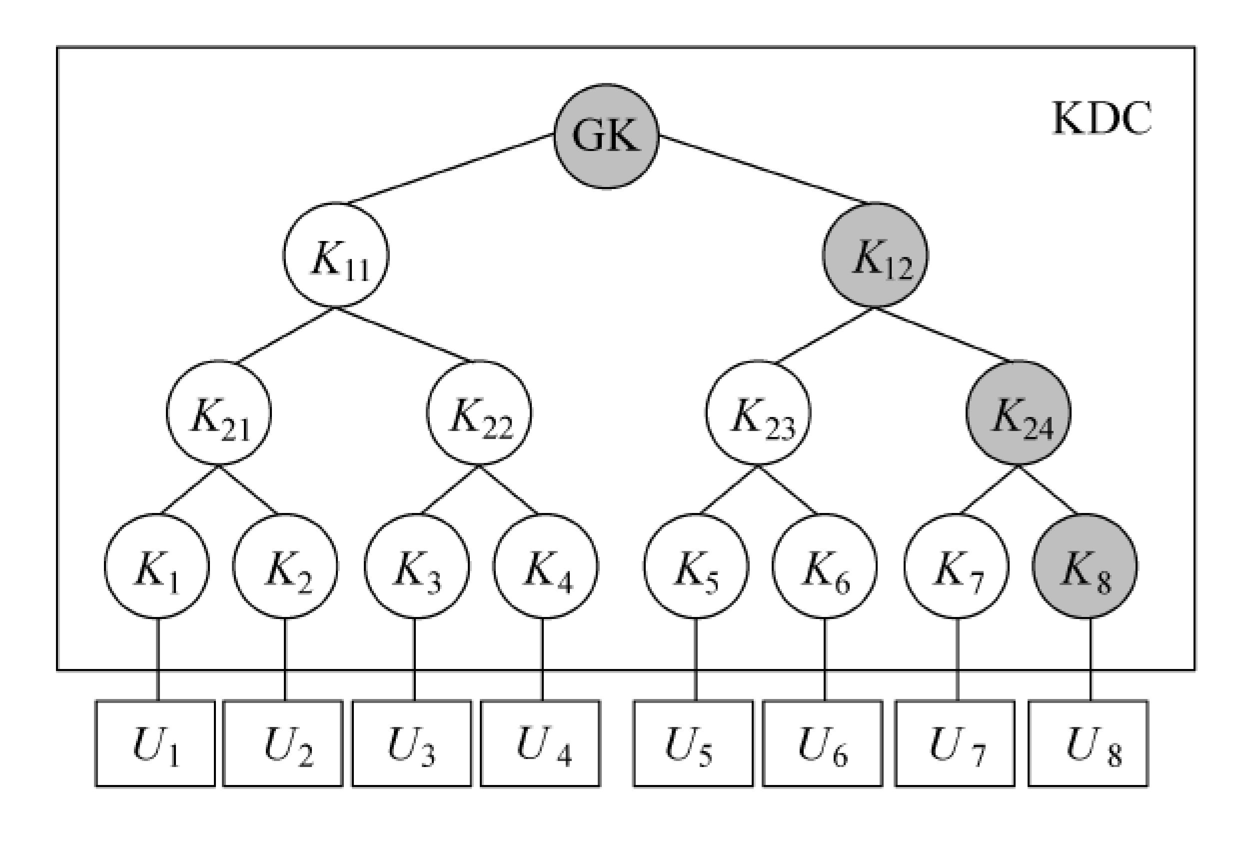


图 4.2.3　LKH 逻辑密钥树

用户节点。圆形节点构成的逻辑密钥树结构保存在 KDC。图中标出了各节点的相关密钥，其中 GK 为组密钥，对应 LKH 树的根节点；三级密钥加密密钥 KEK 分别为 $\{K_{11}, K_{12}\}$，$\{K_{21}, K_{22}, K_{23}, K_{24}\}$，$\{K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_8\}$。各用户节点保存从其父节点到 LKH 树根节点的路径上的所有密钥，如图 4.2.3 所示，用户 U_8 保存的密钥有 $\{K_8, K_{24}, K_{12}, \mathrm{GK}\}$。

密钥更新：当用户成员关系发生变化时，如新成员加入、成员退出，需要进行组密钥更新以保证前向/后向隐私性。

如图 4.2.4(a)所示，当用户 U_8 加入通信组时，则从其父节点到 root 根节点的所有相关的密钥都需要更新以保证后向隐私性。密钥的更新由 KDC 产生，更新过程如下：

(1) 新用户 U_8 加入，KDC 单播一个私有密钥 K_8 给 U_8；

(2) 更新 K_{24} 为 K'_{24} 并用 K_7 和 K_8 加密、更新 K_{12} 为 K'_{12} 并用 K_{23} 和 K'_{24} 加密、更新 GK 为 GK′，用 K_{11} 和 K'_{12} 加密，由这 3 部分共同构成基本组通信密钥更新消息报文：

$$[(K'_{24})K_7][(K'_{24})K_8][(K'_{12})K_{23}][(K'_{12})K_{24}][(\mathrm{GK})K_{11}][(\mathrm{GK})K'_{12}]$$

(3) 组通信密钥更新报文，各用户节点接收并依次解密，获取更新密钥。

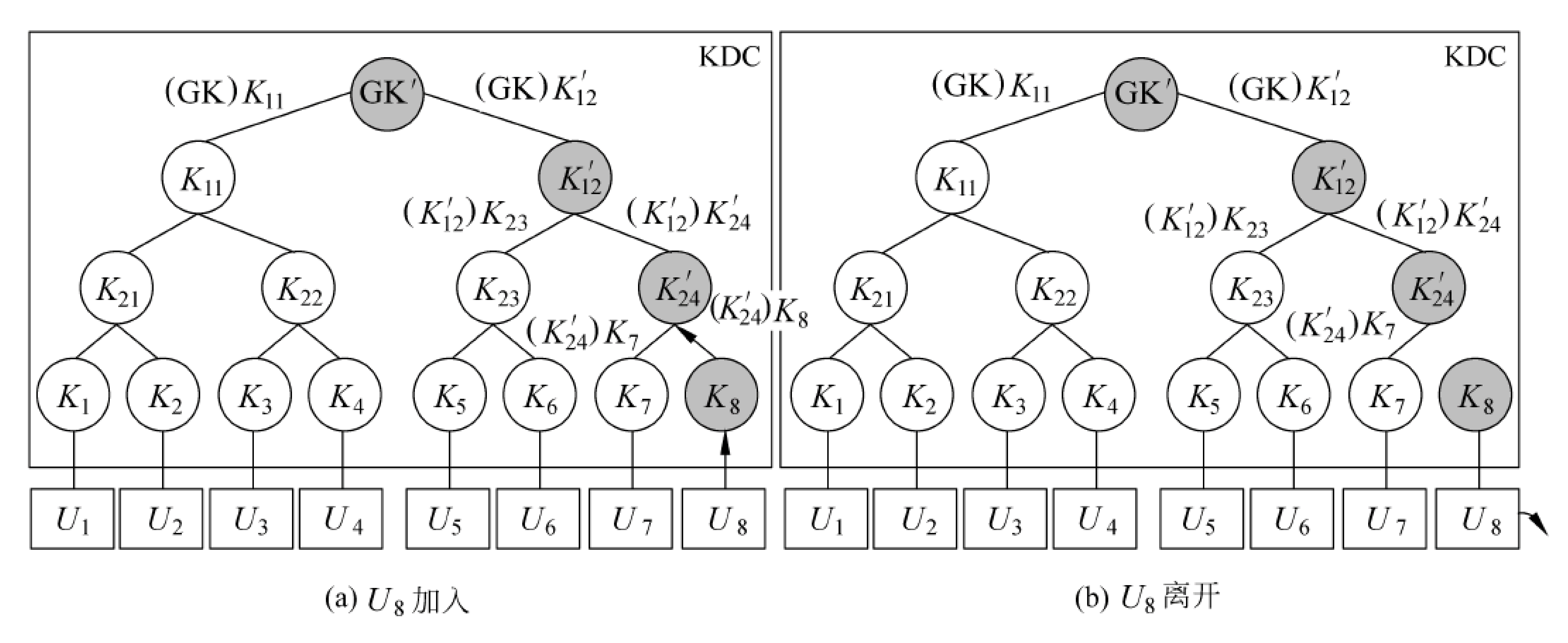


图 4.2.4　用户 U_8 加入/离开密钥更新

如图 4.2.4(b)所示，当用户 U_8 离开通信组，则从其父节点到 root 根节点的所有相关的密钥都需要更新以保证前向隐私性。更新过程与用户加入的更新过程类似。组通信密钥更新消息报文为

$$[(K'_{24})K_7][(K'_{12})K_{23}][(K'_{12})K_{24}][(\mathrm{GK})K_{11}][(\mathrm{GK})K'_{12}]$$

组通信密钥更新报文，各用户节点接收并依次解密，获取更新密钥。

2. 引入单向函数的逻辑密钥树管理

基于单向函数的逻辑密钥树管理[23]在 LKH 树的基础上，通过引入单向函数和混合函数运算，由用户节点计算得到密钥树中 KEK 和 GK，减小密钥更新消息报文长度，提高密钥消息传输效率。该类管理方案是用计算开销换取传输效率的提高，为数据的传输节省了网络带宽。

文献[18,24]提出的集中式单向函数树(centralized one-way function tree，C-OFT)管理方案中引入了如下函数运算：

$$K_i = f(g(K_{\text{left}(i)}), g(K_{\text{right}(i)})) = f(\text{BK}_{\text{left}(i)}, \text{BK}_{\text{right}(i)}) \quad (4.2.1)$$

left(i)和 right(i)分别表示非用户节点 i 的左孩子和右孩子。g 是单向函数，f 是混合函数。通过使用单向函数和混合函数，可以将密钥更新消息的长度从 LKH 的 2(log$n$)降低到 log$n$。从用户节点到根节点的路径中所有节点的集合构成祖先集，祖先集中节点的兄弟构成的集合为兄弟集，各节点保存的密钥包括其私有密钥及其兄弟集中各成员密钥对应的盲钥(blind key，即密钥经过单向函数 g 运算后所得的密钥)。

如图 4.2.5 所示，当用户节点 U_8 加入时进行密钥更新：

(1) KDC 首先向它单播密钥{K_8, BK_7, BK_{23}, BK_{11}}；

(2) 然后，组播经过加密的密钥信息{(BK_8)K_7, (BK_{24})K_{23}, (BK_{12})K_{11}}；

(3) 各节点收到加密的盲钥信息后，先解密出相关的盲钥信息，然后根据自己掌握的其他相关盲钥进行混合运算求得各更新密钥。

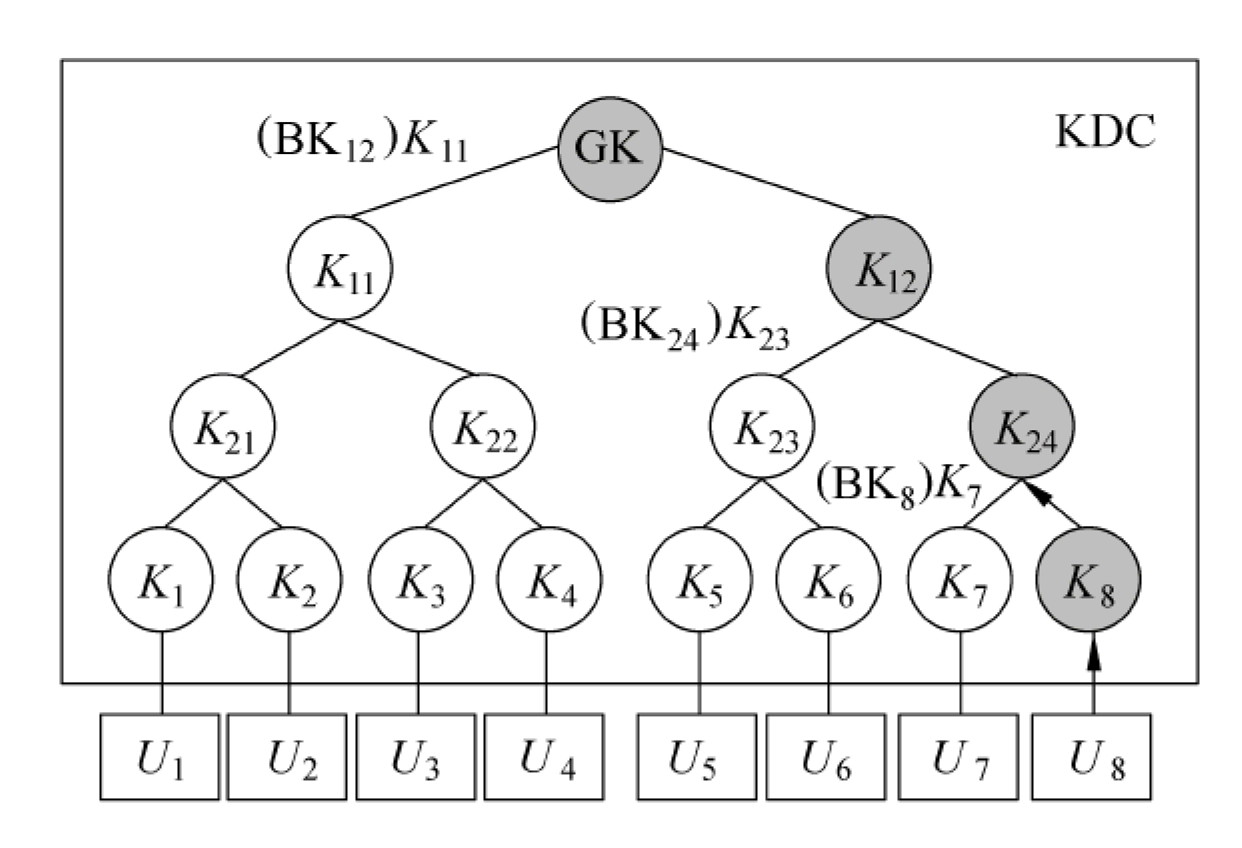


图 4.2.5 集中式单向函数树(C-OFT)

文献[23]提出的 OFCT(one-way function chain tree)管理方案主要针对用户离开时的密钥更新。OFCT 采用伪随机数生成器[25](pseudo random generator)$G(x)$，它可以分为左、右两部分，分别为 $L(x)$和 $R(x)$。$L(x)$, $R(x)$, x 长度相等，即 $G(x)=L(x)R(x)$, $|L(x)|=|R(x)|=|x|$。其密钥更新的方式是：

(1) 当用户 U 离开时，KDC 为从该用户节点到根节点的路径上的每个节点 v 赋一个新值：如果 v 为叶子节点，则其父节点赋值为 $r_{\text{parent}(v)}=r$；否则，其父节点赋值为 $r_{\text{parent}(v)}=R(r_v)$，parent($v$)表示 v 的父节点；

(2) KDC 组通信发送($r_{\text{parent}(v)}$)$K_{\text{sibling}(v)}$，($r_{\text{parent}(v)}$)$K_{\text{sibling}(v)}$ 表示用 $K_{\text{sibling}(v)}$ 加密 $r_{\text{parent}(v)}$，$K_{\text{sibling}(v)}$ 是 v 节点的兄弟节点的私有密钥；

(3) 相关用户收到 r_v，通过 L 函数求得新密钥 $K'_v=L(r_v)$。

如图 4.2.6 所示，当用户节点 U_8 离开时进行密钥更新：

(1) KDC 为节点赋值：$r \to K_{24}$, $R(r) \to K_{12}$, $(R(r)) \to$ Root；

(2) 然后 KDC 组播：{$(R(R(r)))K_{11}$, $(R(r))K_{23}$, $(r)K_7$}；

(3) 各用户节点计算新密钥：用户 U_7 计算 $K'_{24}=L(r)$；

(4) 用户 U_5, U_6, U_7 计算 $K'_{12}=L(R(r))$；用户 U_1, U_2, U_3, U_4, U_5, U_6, U_7 计算 GK′=

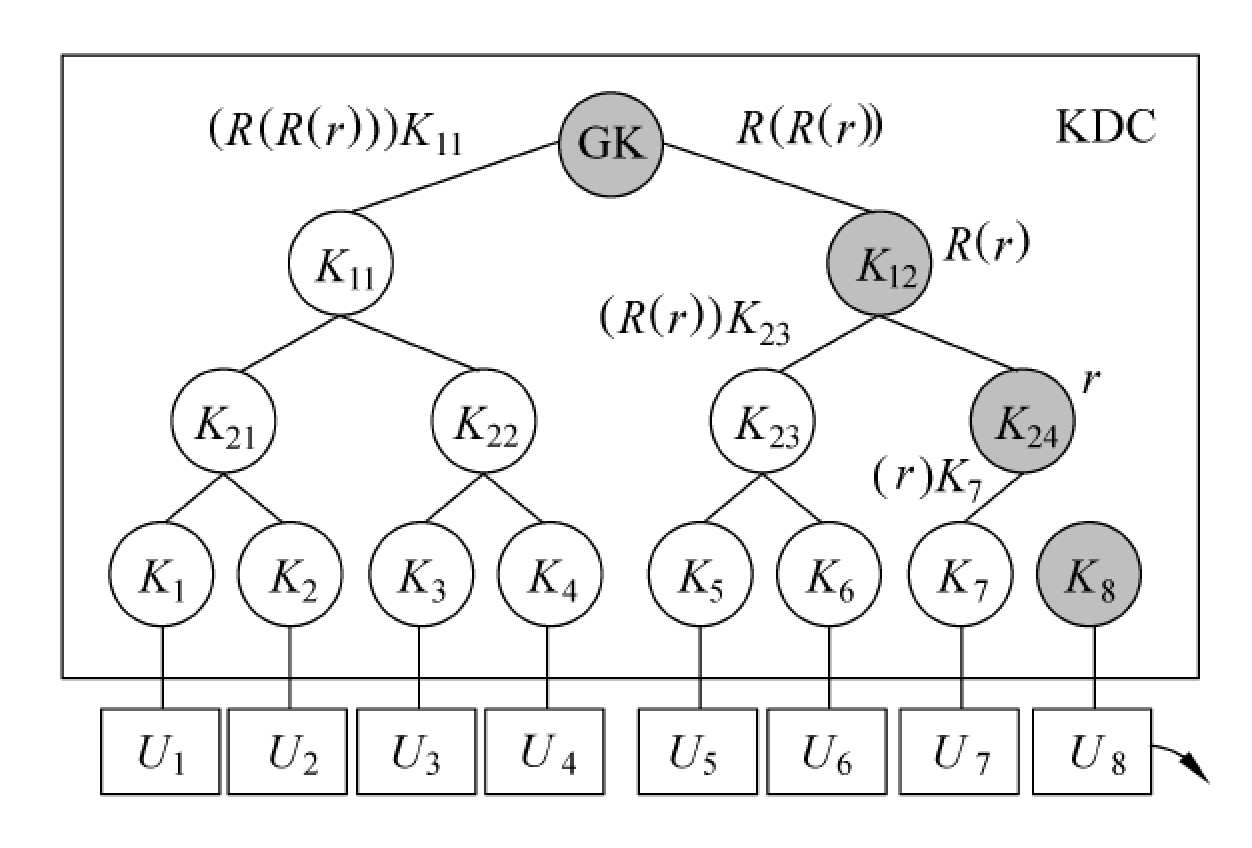


图 4.2.6　单向函数链树 OFCT

$L(R(R(r)))$。

文献[3]提出的 ELK 管理也是基于 LKH 管理方式的，并使用单向函数，管理方式与 C-OFT（单向函数树）方式相近。该方案中使用伪随机函数 PRF（pseudo random function）建立和管理密钥树。另外，该方式提供了密钥恢复机制，在密钥信息丢失的情况下恢复密钥。

3. *a*-ary 树系列管理方案

如图 4.2.7 所示，a-ary 树[26]将 LKH 二叉逻辑树扩展到 a 叉逻辑树，其密钥管理方式与基本 LKH 相似。a-ary 簇树管理方案是在 a-ary 树管理的基础上引入簇（cluster）的管理。方案中，将 n 个通信组用户分成 m 大小的簇，共 n/m 簇。簇中的所有用户共享同一簇 KEK，同时每个用户分别与 KDC 分享一私有密钥 K_i。各簇分别拥有一个随机种子 r，KDC 使用 r 生成 K_i。如图 4.2.8 所示，节点度为 4，簇大小为 3 的 4-ary 簇树。用户离开时，KDC 先用 K_i 加密更新簇 KEK（$m-1$ 次加密操作），然后按照基本的 LKH 方案更新簇节点密钥。

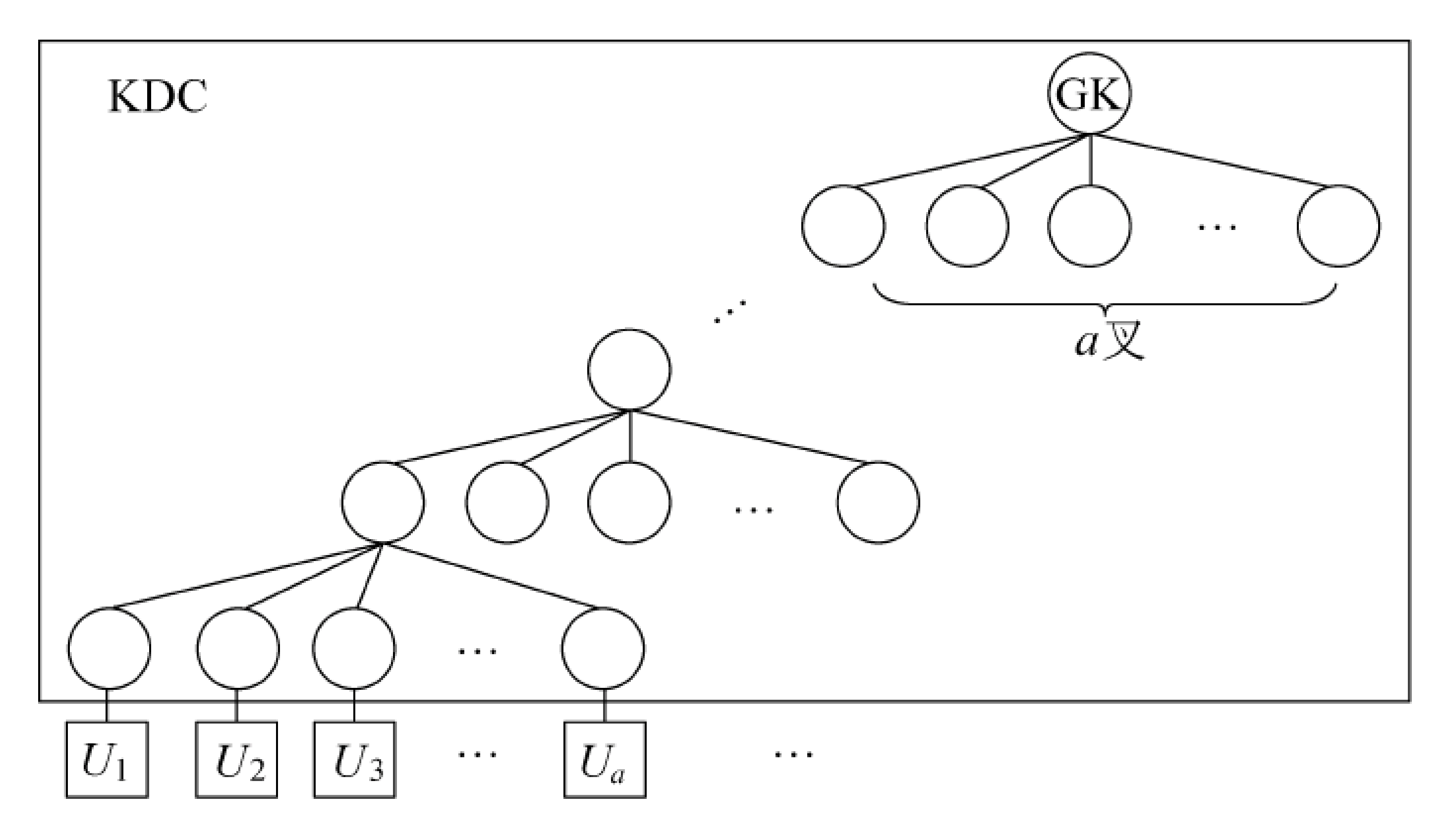


图 4.2.7　a-ary 树

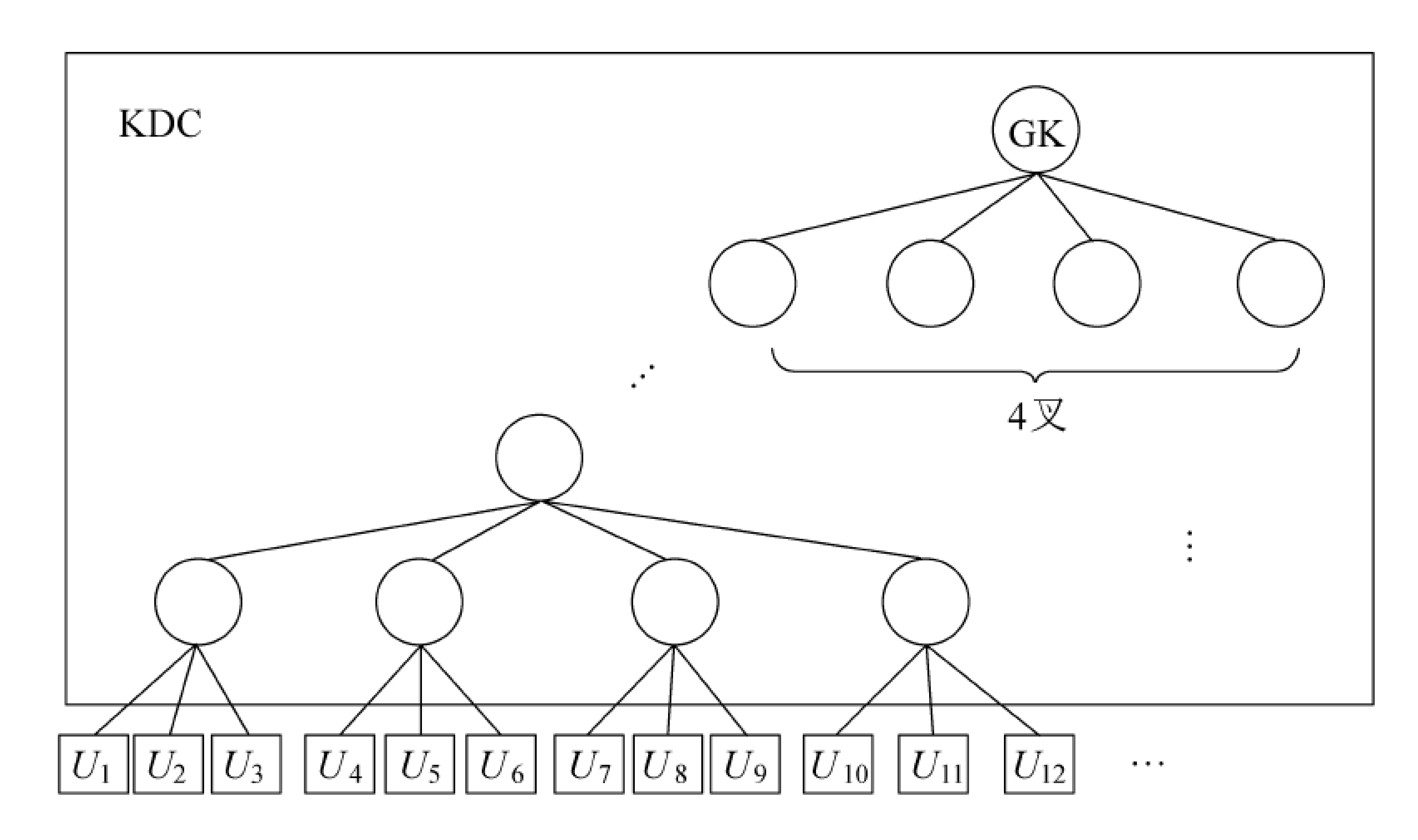


图 4.2.8　簇大小为 3 的 4-ary 簇树

4.2.4.3　典型方案比较

如表 4.2.1 所示，表中对典型的集中式组通信密钥管理方案的管理特点和安全特性进行了比较；表 4.2.2 对各典型集中式组通信密钥管理方案的密钥更新消息的大小进行了比较，表明了各方案的高效性。

表 4.2.1　典型集中式密钥管理方案的特点与安全性比较

	管理特点						安全性		
	平面管理	层次管理	基于树	使用单向函数	可扩展性	密钥恢复机制	前向保密	后向保密	抗同谋破解
GCP	Y	N	N	N	N	N	Y	N	Y
C-FT	Y	N	N	N	Y	N	Y	Y	N
LKH	N	Y	Y	N	Y	N	Y	Y	Y
C-OFT	N	Y	Y	Y	Y	N	Y	Y	Y
OFCT	N	Y	Y	Y	Y	N	Y	Y	Y
ELK	N	Y	Y	Y	Y	Y	Y	Y	Y
a-ary	N	Y	Y	N	Y	N	Y	Y	Y
a-ary cluster	N	Y	Y	N	Y	N	Y	Y	Y

表 4.2.2　典型集中式密钥管理方案高效性比较(密钥更新消息大小)

	用户加入(组播)	用户加入(单播)	用户离开(组播)
GCP	2	2	—
C-FT	$2\log n$	$\log n+1$	$2\log n$
LKH	$2h-1$	$h+1$	$2h$
C-OFT	$h+1$	$h+1$	$h+1$

续表

	用户加入(组播)	用户加入(单播)	用户离开(组播)
OFCT	h	$h+1$	$h+1$
ELK	0	$h+1$	n_1+n_2
a-ary	$2\log n-1$	$\log n+1$	$a\log n$
a-ary cluster	$m-1+a\log n/m$	$\log n/m+2$	$m-1+a\log n/m$

4.2.5　分散式组密钥管理

4.2.5.1　密钥分发服务器模式

1. 可扩展组通信密钥分发

可扩展组通信密钥分发(scalable multicast key distribution,SMKD)[27]使用 CBT(core based tree)组播路由协议建立密钥树,用来在通信组内分发密钥。从加入用户到核心路由器的整个路径上的路由器均可以对用户进行身份认证并可以分发组密钥,而这些路由器由核心路由器进行认证。这种模式明显提高了安全控制的可扩展性。其主要缺陷在于:首先,这种模式需要对 IGMP 协议进行修改,并要采用 CBT 组播路由协议;其次,缺乏有效的前向隐私性保障,当用户离开通信组时,保证前向隐私的唯一途径就是重建安全通信组;第三,因为 CBT 中的路由器均掌握组播组密钥,所以 SMKD 需要 CBT 中的路由器是高度可信任的。

2. 域内组密钥管理

如图 4.2.9 所示,域内组密钥管理(intra-domain group key management,IGCP)[28]将通信组分为若干区域(area)。通信组内具有一个域级密钥分发器 DKD(domain-wide key distributor)。每个区域有一个区域密钥分发器 AKD(area key distributor)。整个域通信组具有唯一的组密钥,组密钥由 DKD 产生,由相关的 AKD 进行分发。由于 IGCP 采用 DKD 进行集中控制,所以 DKD 成为单一失效点,当 DKD 发生故障时,则导致整个通信组无法工作。

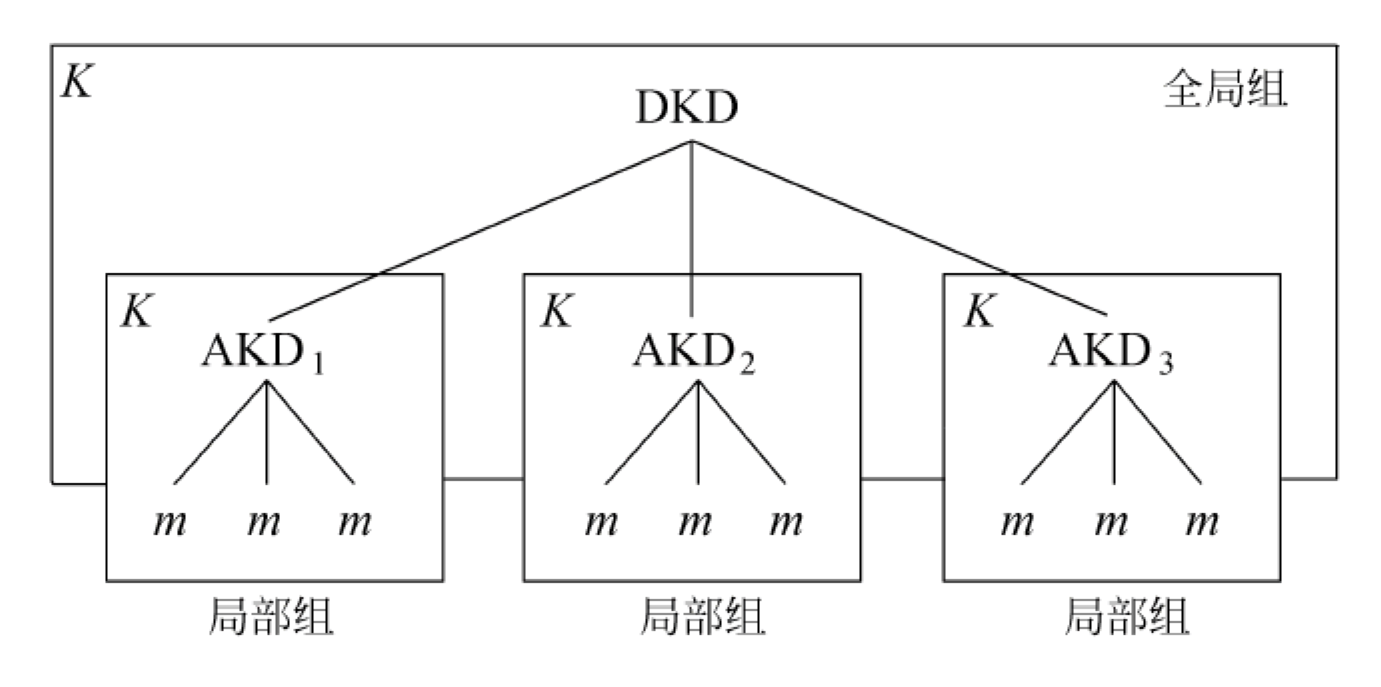


图 4.2.9　域内组密钥管理

3. 双重加密协议

双重加密协议(dual-encryption protocol,DEP)[29]通过双重加密解决了其他模式中第三方(子组控制器)必须可信的问题,通过双重加密实现在子组管理器 SGM(subgroup manager)不可完全信任的情况下,实现安全组通信。该模式中,由子组管理器控制管理各相应的子组。具有 3 种密钥加密密钥(KEK)和一个数据加密密钥 DEK(data encryption key)。KEK_{i1} 由 SGM_i 及其组成员持有,KEK_{i2} 由 GC 与 i 子组除 SGM_i 之外的组成员持有,KEK_{i3} 由 SGM_i 和 GC 共享。

GC 分发 DEK 时,先将 DEK 用 KEK_{i2} 加密,再用 KEK_{i3} 加密,当 SGM_i 收到消息,用 KEK_{i3} 解密,然后用 KEK_{i1} 加密在本地子组内分发,子组用户分别依次用 KEK_{i1} 和 KEK_{i2} 解密,获取 DEK。在整个过程中,作为第三方的 SGM_i 不必知晓 DEK,所以无法解密得到原始数据,不需要具有高度的信任关系。当用户退出,用户与子组管理器间共享的密钥更新,而数据加密密钥未更新时,退出用户仍能访问组内数据。这成为 DEP 在前向保密性方面存在的缺陷。

4. Hydra

Hydra[30]是一种具有良好的可扩展性、适用于大规模通信组的分散式密钥管理模式。该模式中,通信组分为若干 TTL(time-to-leave)区域,每个区域由指定的 Hydra 服务器管理控制(hydra server,HS)。Hydra 不采用集中控制器对 Hydra 服务器进行管理,而是采用 HS 间协同合作的方式,这样管理更灵活,避免了集中管理的单失效点造成的系统瓶颈。为使 HS 之间同步工作,采用同步组密钥分发协议 SGKDP(synchronized group key distribution protocol)。Hydra 中采用唯一的组密钥,当密钥更新时,通过 SGKDP 协议,使所有 HS 服务器认可新的组密钥。

5. Kronos

Kronos[31]采用唯一的域级数据加密密钥,用定期密钥更新替代事件驱动的密钥更新,组密钥在一定时间周期内产生。在当前周期内,对所有组成员的变化均进行收集,在当前周期将结束、新组密钥分发时进行处理。Kronos 方案将通信组分为若干区域,每个区域有一个区域密钥分发器。此外,具有一个域级密钥分发器。所有 AKD 共同认可两个密钥因子:K,R_0,可以通过安全信道从 DKD 获得。各 AKD 在 K,R_0 基础上定期独立生成相同的域级数据加密密钥(domain-wide data encryption key)。

所有 AKD 使用 NTP (network time protocol)进行时钟同步,保证不同 AKD 进行的定期密钥更新是同步的。各 AKD 生成域级数据加密密钥的过程:$R_1 = E_K(R_0)$,$R_{i+1} = E_K(R_i)$,其中 E 为对称加密操作。Kronos 方案无集中控制器,域级数据加密密钥由各 AKD 自行独立产生,容错性强。其缺陷在于:在密钥因子的基础上产生密钥,密钥被破坏的可能性加大,安全性变差。

4.2.5.2 重加密服务器模式

1. IOLUS

在集中式组通信密钥管理方案中,由于集中控制的存在,中央控制器容易成为系统的瓶颈。中央控制器的故障会导致整个安全组通信系统所依赖的唯一组密钥生成和分发中断,

发生 1-affect-n 问题。要有效地提高系统的鲁棒性，可以采用 IOLUS[32] 管理方案。

如图 4.2.10 所示，IOLUS 将通信组成员分为若干子通信组，子通信组由组安全中间节点 GSI(group security intermediary)控制构建自治域，由 GSI 控制在子通信组范围内进行密钥管理；各 GSI 与组安全控制中心 GSC(group security controller)构建上一级的通信组，由 GSC 控制构建自治域，管理各 GSI 的密钥更新、创建和分发。随着用户的增加，可以构建下一级的自治域，形成一种分布安全树结构。如图 4.2.10 所示。G^1，G^{2C}，G^{2B}，G^{2A}，G^{3B}，G^{3A} 分别构成自治域，其中 G^1 为顶级通信组，G^{2C}，G^{2B}，G^{2A} 为一级通信组，G^{3B}，G^{3A} 为二级通信组，依次类推。

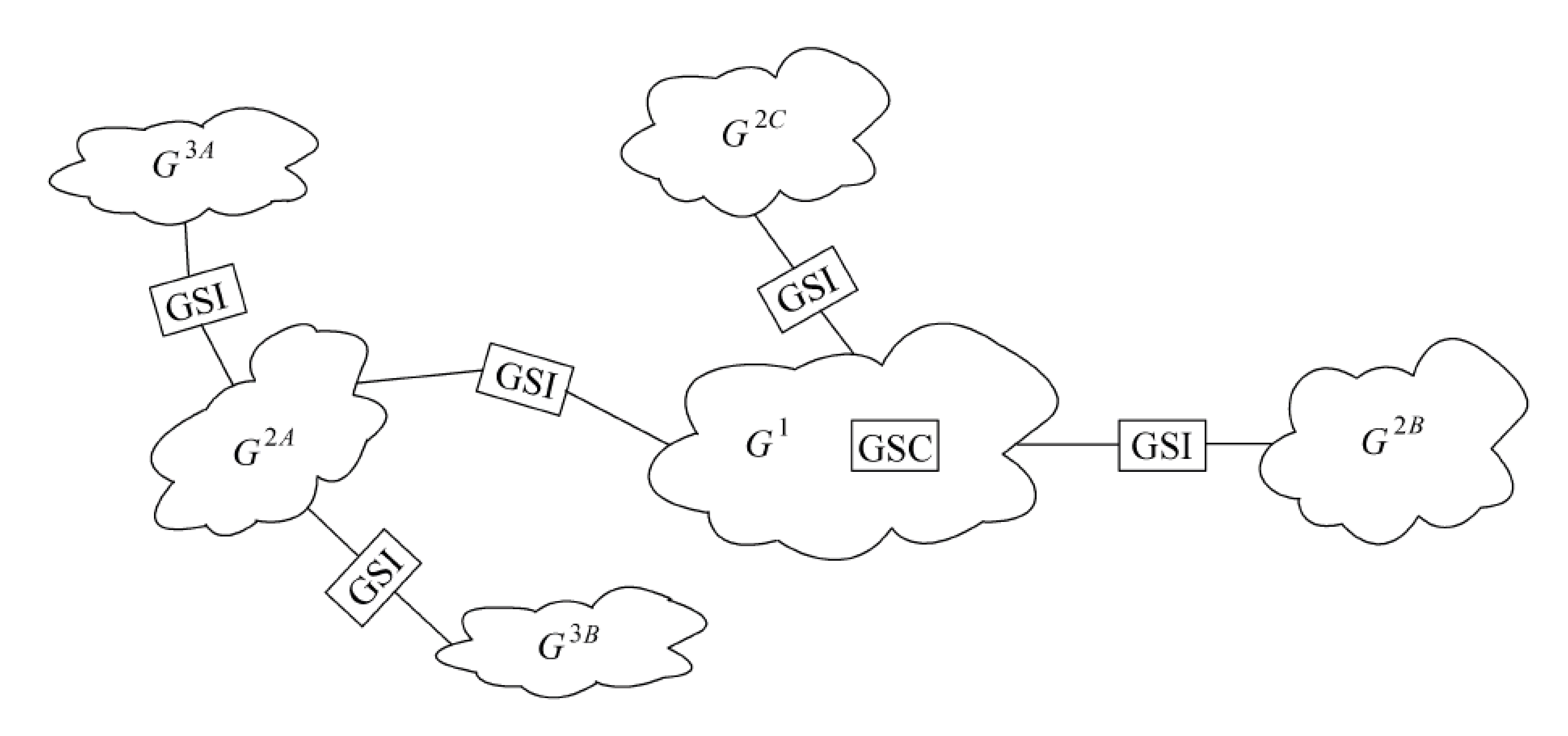


图 4.2.10 IOLUS 管理方案

基于这种分级分层的控制方式，GSC 只需管理各一级 GSI 的密钥更新，可以有效地减少 GSC 负载，降低集中控制方式中中心控制节点的脆弱性，提高系统的可靠性。各自治域在各自控制节点的管理下，在本自治域的范围内进行密钥管理(密钥更新消息不需要在整个通信组内更新，只需在本子通信组内更新)，减小密钥更新的时间延迟，提高了密钥更新的有效性。各级的密钥更新均可以采用 LKH，LKH＋，C-OFT 等集中式控制模式。

IOLUS 中的数据传输通过 GSI 的交替转发(relay)实现。如图 4.2.11 所示，数据传输方式分为直接组播(direct multicasting)和 GSA 辅助组播(GSA-assisted multicasting)两种。在直接组播方式中，发送方通过用子组密钥 K_{SGRP} 将数据加密，直接在本地子组内组播数据，完成本地子组数据发送；本组 GSI 接收到数据，首先将数据密文解密，并用上层组密钥加密，然后进行跨组转发发送。在 GSA 辅助组播方式中，发送方用私钥 $K_{GSA\text{-}MBR}$ 加密数据，单播给 GSI，GSI 解密，然后分别用本组组密钥和上层组密钥加密，分别完成本地和跨组数据发送。采用这种方式可以避免出现子组密钥 $K_{GSA\text{-}MBR}$ 过期问题。

IOLUS 的主要缺陷在于，GSI 参与数据传输中进行数据加解密处理和数据转发，开销较大，使数据传输路径发生改变，容易形成系统瓶颈；另外，IOLUS 中要求所有 GSI 具有完全信任关系。

2. 加密序列

加密序列模式 CS(cipher sequences)[33] 采用非对称式加密方式，各中间节点采用不同的密钥进行加密操作，中间节点无需解密出原始数据，也不需要具有完全信任关系，适用于

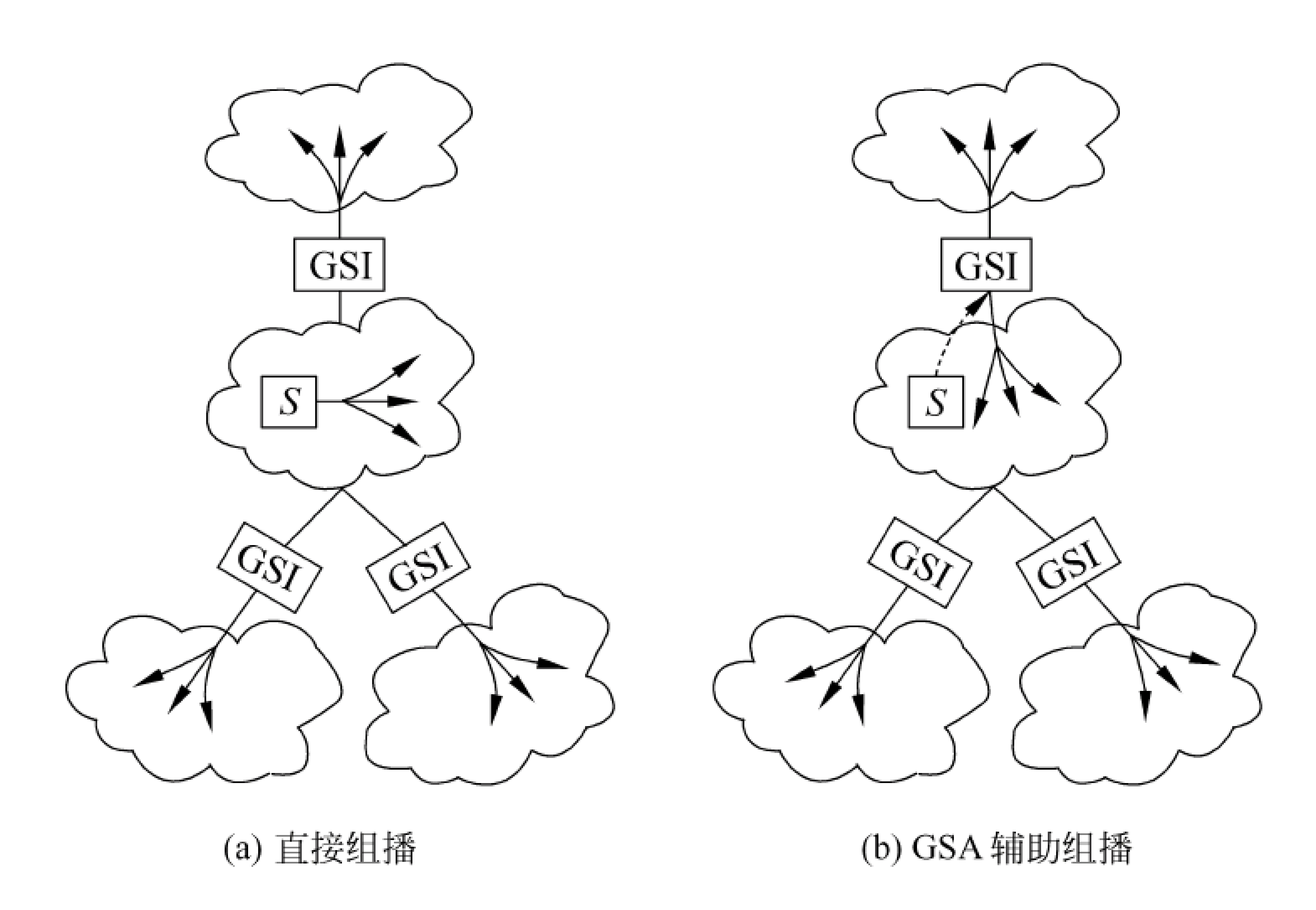


图 4.2.11　IOLUS 数据传输方式

小规模组数据的加密传输。

CS 方案中引入加密组(cipher group)概念。函数 $f(S,a)$ 为加密组，如果两个元素序列：a_i($1\leqslant i\leqslant n$)和 S_i($0\leqslant i\leqslant n$，S_0 为初始值)，有 $S_i=f(S_{i-1},a_i)$，且对于(i,j)，$i<j$，则有函数 $h_{i,j}()$满足 $S_i=h_{i,j}(S_j)$。

加密序列模式采用树结构，S_0 为要组播的信息。如图 4.2.12 所示，每个圆形非叶子节点 N_i 分配一个 $a_i>1$，各节点 N_i 可以提供 f 函数运算。方形叶子节点对应 h 函数：$S_0=h_{0,n}(S_n)$。原始数据信息经过从根节点到用户节点路径上中间节点的逐次加密运算 $f(S_{i-1},a_i)$，在用户节点可以通过 $S_0=h_{0,n}(S_n)$还原初原始数据。特点在于采用非对称式加密操作，各中间节点进行加密时采用的密钥均不同，中间节点无需知道原始数据，也不需要完全信任支持。缺陷在于限于非对称加密方式，只适用于小规模数据的加密传输。

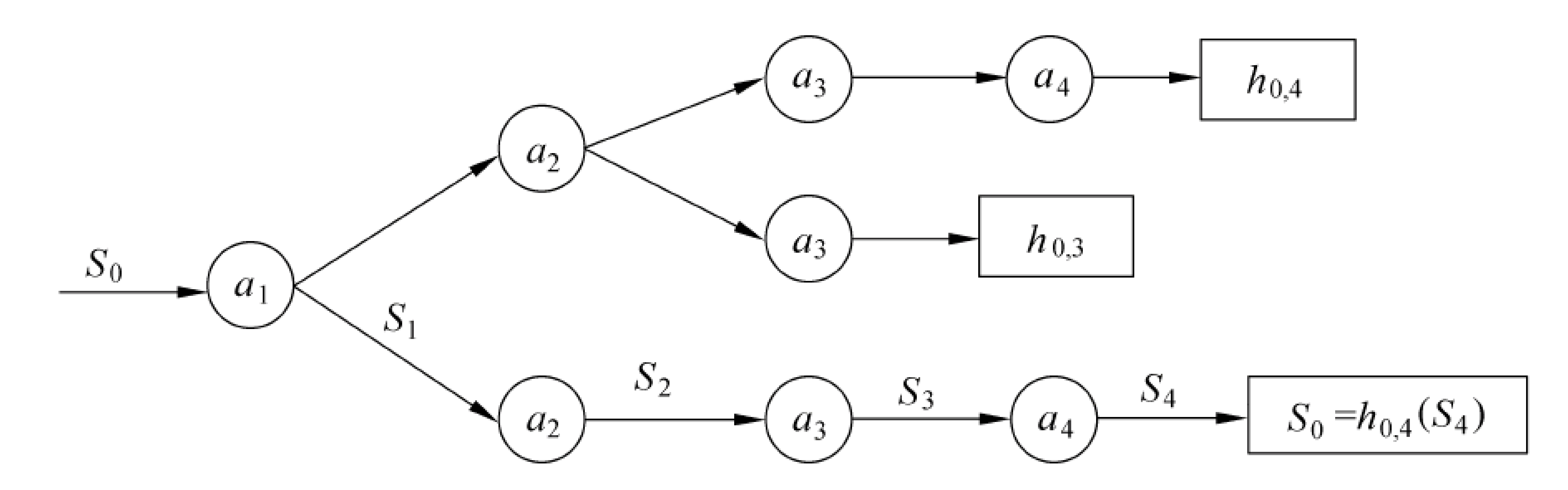


图 4.2.12　加密序列(cipher sequence)

综上所述，现有密钥管理方案所能保证的特性存在着很大的差异性。很多方案在兼顾多方面特性的基础上，着重保证某方面的特性。如定期/批处理密钥更新虽然能够提供高效的密钥网络更新效率，但却牺牲了一定程度的安全性；双重加密协议无需中间控制器完全可信任，但需要通过增加通信密钥数量和加密/解密计算开销来达到该目标。所以，在安全组通信中，确定密钥管理方案应遵从应用的特点加以考虑，寻求最佳管理方案。

4.2.6　分布式组密钥管理

4.2.6.1　非协商模式

分布式 LKH 管理 D-LKH(distributed LKH)[32]取消中央控制器,没有任何实体能获知所有的密钥,没有子组;不同子树之间认同相互之间的通信密钥。

分布式单向函数树管理 DISEC[34]是将单向函数树管理方式拓展到分布式控制中,取消集中控制器。每个组用户具有完全的信任关系,可以进行访问控制和密钥生成。用户负责生成自身的密钥并发送给它的兄弟节点。下一节我们将提出一种新的基于单向函数的分布式密钥管理方案,并与 DISEC 进行性能比较。

VersaKey[19]采用分布式平面表管理方式(distributed flat table,D-FT),通信组中没有指定的组控制器。每个组用户都掌握与其相关的 KEK,都可以完成接纳控制管理和其他管理功能。此外,由于这些管理功能很容易迁移到其他节点,所以这种管理方式具有良好的适应性,可以克服节点故障,具有自恢复性。其缺陷在于,不提供密钥协商机制,用户加入时需要和一组用户联系才可以获知它所需要的所有密钥,当多个用户同时更新同一密钥时,同步操作需要相当的时间延迟。此外,该方式容易受到同谋破解攻击。

4.2.6.2　协商模式

1. CLIQUES

在基于 DH 算法的密钥协商机制中,组通信中所用密钥的生成不是由中心控制服务器生成的,而是通过通信组成员共同协商建立,使得在通信组密钥生成、更新过程中保持了公平性。CLIQUES[35,36]密钥管理方案利用 Diffie-Hellman 密钥协商算法[37]的扩展来实现组密钥的协商生成。

在 CLIQUES 初始化中,具有 n 个组成员的通信组通过 n 轮的协商获得组密钥 GK(图 4.2.13 为 4 用户初始化的情况):

(1) 确定所有组成员认可的 DH 参数:q,g,然后各自选定一个随机数 N_i,并计算幂值 g^{N_i};

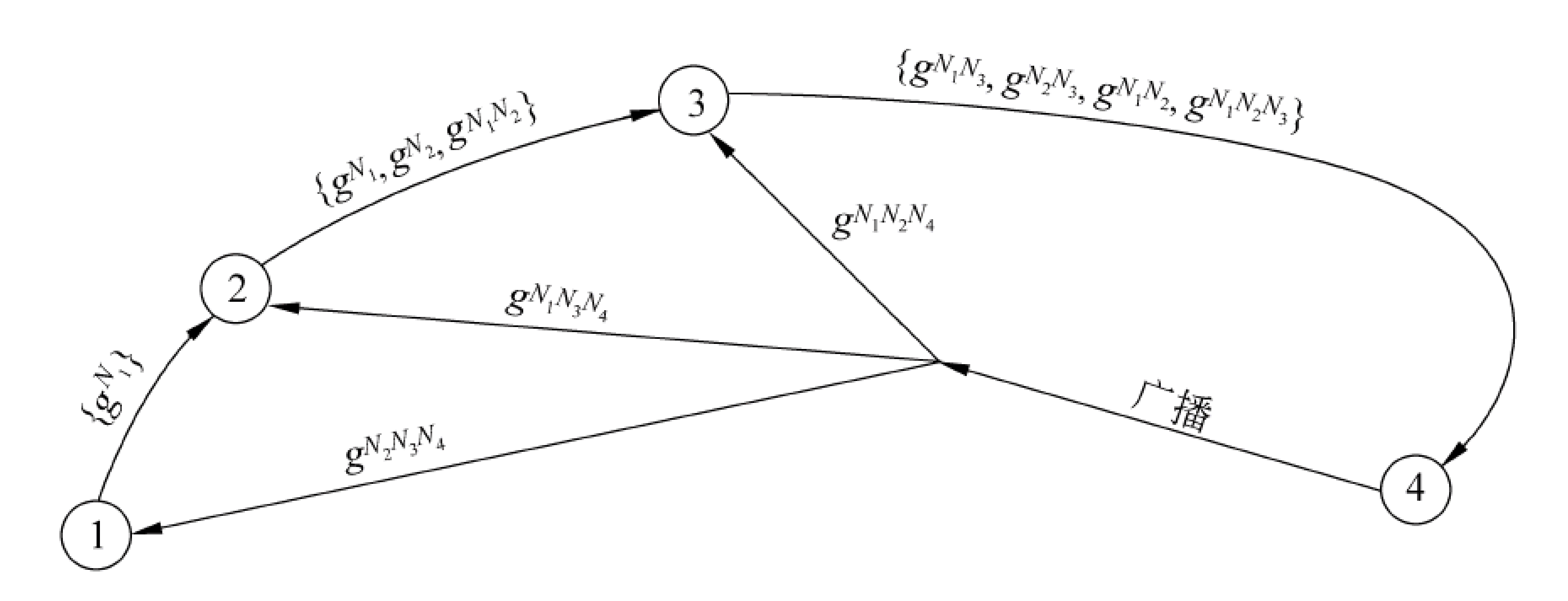


图 4.2.13　4 用户的 CLIQUES 初始化

(2) 进行 $n-1$ 轮协商：第 1 个用户 U_1 计算 $S_1=\{g^{N_1}\}$，并发送给第 2 个用户 U_2；第 2 个用户 U_2 计算 $S_2=\{g^{N_1},g^{N_2},g^{N_1N_2}\}$，并发送给第 3 个用户 U_3；依次类推，第 $i\in[1,n-1]$个用户 U_i，计算生成 S_i 并发送给用户 U_{i+1}：$S_i=\{g^{N_1N_2\cdots N_i/N_k}\mid k\in[1,i]\}\cup\{g^{N_1N_2\cdots N_i}\}$；

(3) 最后一个组成员 U_n 收到 S_{n-1}，计算 $S_n=\{g^{N_1N_2\cdots N_n/N_k}\mid k\in[1,n-1]\}$，并向其他 $n-1$ 个组成员广播，各组成员收到并计算 $\mathrm{GK}=g^{N_1N_2\cdots N_n}\bmod q$。

当有新用户 U_{n+1} 加入通信组时，用户 U_n 首先更新 N_n 为 $\overline{N}_n$，然后计算并发送密钥基值 $S_n=\{g^{N_1N_2\cdots\overline{N}_n/N_k}\mid k\in[1,n]\}\cup\{g^{N_1N_2\cdots\overline{N}_n}\}$ 给新用户 U_{n+1}，然后，新用户 U_{n+1} 收到 S_n，计算 $S_{n+1}=\{g^{N_1N_2\cdots\overline{N}_nN_{n+1}/N_k}\mid k\in[1,n]\}$，并向其他用户进行广播；各组成员收到并计算新的 $\mathrm{GK}=g^{N_1N_2\cdots\overline{N}_nN_{n+1}}\bmod q$。如图 4.2.14 所示，为用户 U_5 加入通信组时密钥更新的情况。

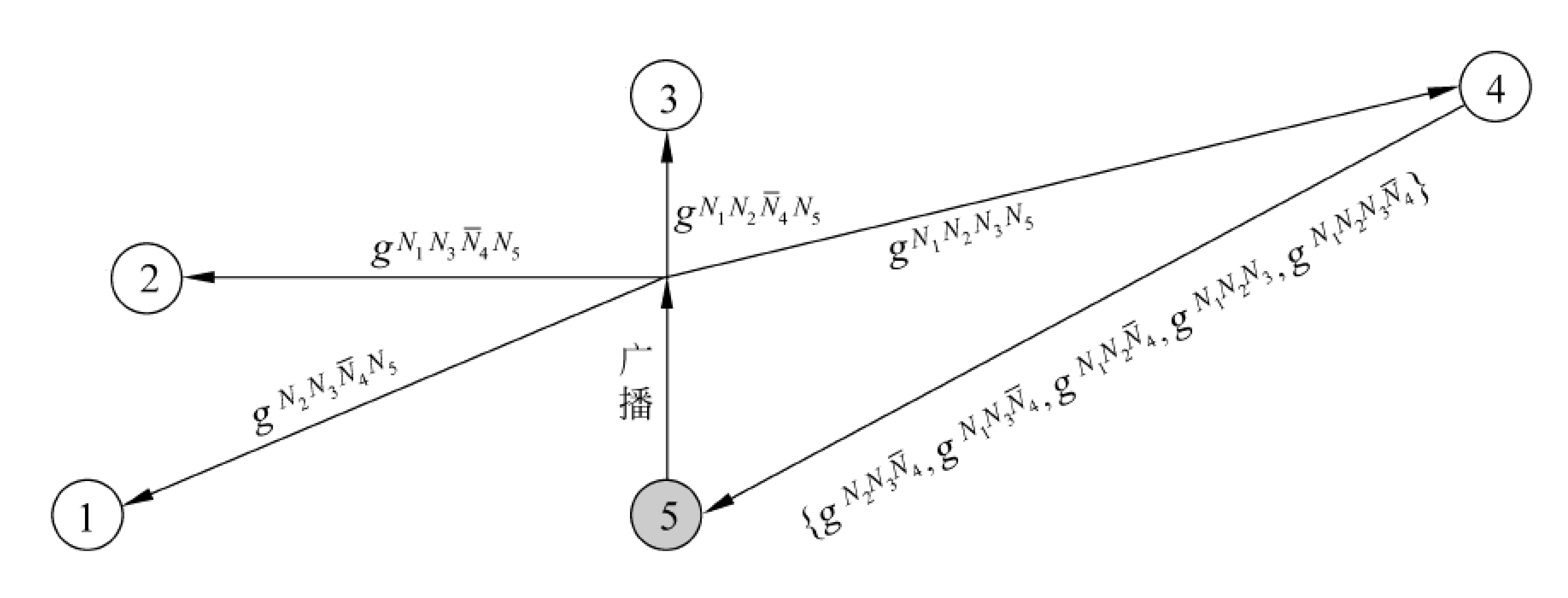


图 4.2.14 用户 U_5 加入通信组的情况

当有用户 U_p 离开通信组时，用户 U_n 计算新的密钥基值并向其他用户进行广播：$S_n=\{g^{N_1N_2\cdots\overline{N}_n/N_k}\mid k\in[1,n-1]\land k\neq p\}$，除用户 U_p 之外的组成员收到密钥基值并计算新的 $\mathrm{GK}=g^{N_1N_2\cdots\overline{N}_n}\bmod q$，如图 4.2.15 所示，为用户 U_3 离开组时的情况。

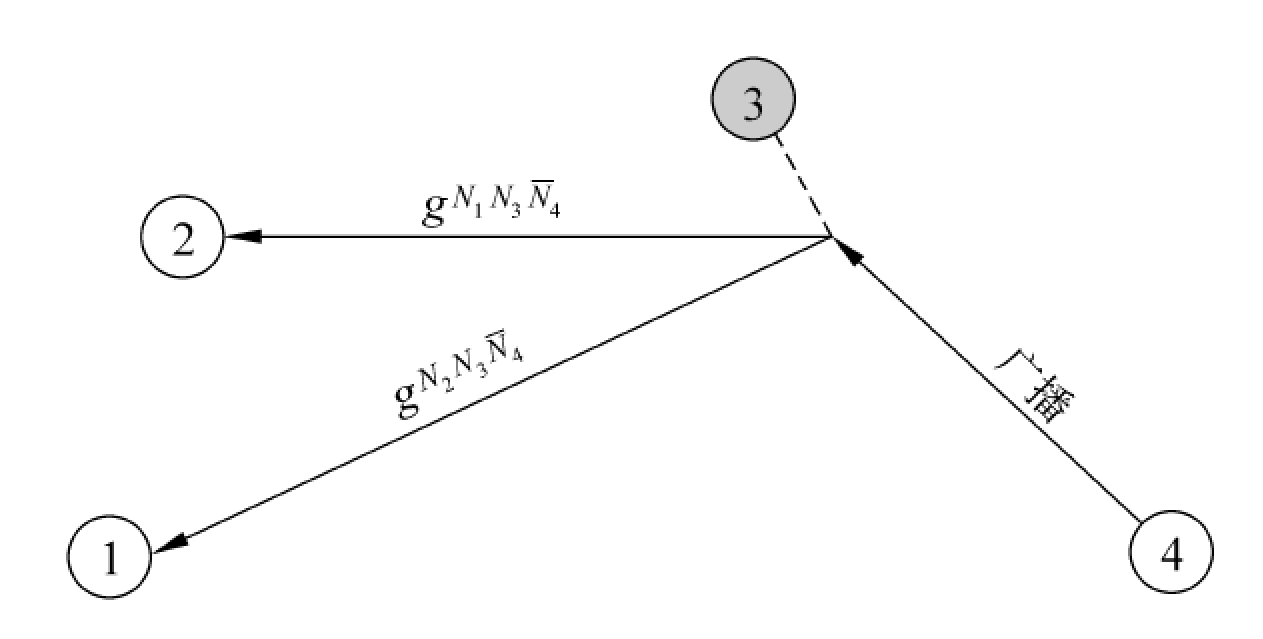


图 4.2.15 用户 U_3 离开通信组的情况

CLIQUE 的密钥传输时间延迟的复杂度为 $O(n)$，密钥计算的总计算量为 $O(n^2)$，密钥传输所占用的带宽为 $O(n^2)$。因此，CLIQUE 的扩展性较差。

2. 基于树的组 DH 协商

基于树的组 DH 协商 TGDH(tree-based group DH key agreement)[38,39] 采用 Diffie-Hellman 算法替代单向函数协商产生高层密钥，可以减少组用户保存的密钥数目。

3. Octopus 协议

Octopus 协议[40]模式也是基于 Diffie-Hellman 协商。该模式中，组用户被分为 4 个子组。每个小组内部进行 DH 协商 $I_{\text{subgroup}}=\alpha^{u_1 u_2 \cdots u_n/4}$，各子组协商完成后，小组间交换协商的密钥基值，进一步协商得到通信组密钥。

4. 会议密钥协商

会议密钥协商(conference key agreement，CKA)[41]通过函数实现所有用户共同参与产生组密钥。该模式指定 $n-1$ 个用户明文广播他们相应的密钥基值，组领导用每个用户相应的公钥加密其提供的密钥基值并广播。持有相应公钥的用户可以解密并使用组领导提供的基值，并计算组密钥为 $GK=f(N_1, h(N_2), \cdots, h(N_n))$，其中 f 为连接函数，h 为单向函数，n 为组成员数目，N_i 为用户 i 提供的密钥基值。

5. BD 协议

Burmester 和 Desmedt 提出 BD 协议(Burmester/Desmedt Protocol，BDP)[42]协议实现高效的密钥协商。协商过程只需要 3 轮：

(1) 用户 U_i 生成随机数 r_i 并广播 $Z_i=\alpha^{r_i}$；

(2) 用户 U_i 计算并广播 $X_i=(Z_{i+1}/Z_{i-1})^{r_i}$；

(3) 用户 U_i 计算通信组密钥 $Z_{i-1}^{nr_i} \cdot X_i^{n-i} \cdot X_{i+1}^{n-2} \cdots X_{i-2} \bmod p$。

6. STR 协议

STR[43]方案简单且具有容错能力，可以较好地保证鲁棒性，以运算的开销换取网络传输效率的提高，适用于长时间延迟的广域网络环境。

7. 典型密钥协商管理方案比较

该类各典型协商方案的比较情况见表 4.2.3。几种典型的协商算法在协商轮数、密钥消息长度、协商运算次数等性能参数均有不同。但相同的是几种方案在进行密钥传输时均只适合采用媒体无关信道方式。这是由密钥通过协商生成且数据源不固定的特点所决定的。

表 4.2.3　多方合作应用模式典型密钥协商管理方案比较

密钥协商管理方案		协商轮次	幂运算次数	鲁棒性保证	消息长度	
					单　播	组　播
STR	用户加入	1	2	容易	1	1
	用户离开	1	$3n/2+2$		0	1
	组分裂	1	$3n/2+2$		0	1
	组合并	2	$3k$		2	1
CLIQUES	用户加入	2	$2n$	难	1	1
	用户离开	1	n		0	1
	组分裂	1	n		0	1
	组合并	$k+3$	$n+2k$		$n+2k-1$	2

续表

密钥协商管理方案		协商轮次	幂运算次数	鲁棒性保证	消息长度	
					单　播	组　播
TGDH	用户加入	2	$\log n$	容易	0	3
	用户离开	1	$\log n$		0	1
	组分裂	$O(\log n)$	$O(\log n)$		0	$O(\log n)$
	组合并	2	$\log n$		0	3
Octopus		$(n-4)/2+2$	$n/4$	难	$3n-4$	0
BDP		3	$n+1$	容易	0	$2n$
CKA		3	—	容易	$n-1$	n

注：n 为通信组用户数；k 为组合并时新加入的用户数。

4.2.7　当前研究热点

4.2.7.1　可靠性研究

对于集中式安全组通信密钥管理方案而言，中央控制节点（组控制器）是单失效节点，它的故障会导致整个通信组的故障。其他两种方案（分散、分布）均非集中式控制，可以有效地避免由于组控制中心节点（GC）负载过重、故障等问题使之成为系统的瓶颈，从而降低系统的脆弱性，但同样具有不易管理等缺点；所以，如何避免各种管理模式中的缺陷，弥补不足，提高组通信系统的可靠性是至关重要的。可否在安全组通信密钥管理的过程中引入延时重发和消息握手机制（密钥更新消息发送后，发送者等待接收者反馈确认信息，特定时间间隔内未收到确认信息，视情况重发密钥更新消息或将无反馈信息的节点作失效节点处理），在密钥分发、更新等过程中确保 re-key 消息可靠传递，或引入密钥恢复机制，以增大密钥更新消息的长度为代价换取密钥更新的可靠性都是非常值得研究的问题。

所以，需要在安全组通信的密钥管理方案中引入密钥恢复机制，保证密钥更新的可靠性。文献[3,9]提出了在不可靠的网络上进行大规模动态组密钥管理的方法。在不可靠的网络上，密钥更新消息可能在传输过程中丢失而不能到达目的节点。这种情况下，用户可以通过请求组控制器进行消息重传，但重传将增加网络带宽开销。在 Keystone[9] 和 ELK[3] 方案中，用户无需与组控制器进行交互，也无需进行附加的重传操作，即可自身恢复密钥。它们通过扩展附加密钥更新信息来防止密钥消息丢失。Keystone 和 ELK 分别采用错误更正编码（error correction code）和线索（hints）来恢复密钥信息。此外，文献[10]提出了一种自恢复密钥分发机制，扩展了文献[11,12]中提出的子集差异密钥更新技术（subset difference re-keying techniques）。文献[10]提供的机制可以保证用户在离线一段时间的情况下，能够在它重新返回时立刻恢复新的通信组密钥。

综上所述，将密钥恢复机制引入，使之与密钥管理方式融合，以提高组通信系统的可靠性研究是有意义的，而且是可行的。

4.2.7.2　与具体网络环境的结合研究

将安全组通信放在具体的网络环境中进行研究，应更多地考虑网络环境的特点，并将组通信密钥管理进行完善，以适应网络环境的要求。

组通信密钥管理的研究可以结合应用层网络的研究进行。将组通信密钥管理机制与应用层组通信的框架结构进行有机的组合，可以提高系统的安全性，且这种方案符合应用层网络的相关特点。

组和组通信的概念是 P2P 系统的基础，前面讨论的分布式组通信密钥管理方式很容易推广到 P2P 应用中。在 P2P 编程框架中，广泛支持组的创建和维护以及组内节点通信[44]。但是，P2P 系统的安全问题还亟待研究和解决。P2P 系统中缺少中央授权机制。附属关系和授权的管理由组内用户共同完成实现。所以，组通信密钥管理的核心问题由密钥分发的高效性转变为组通信密钥的协商建立。此外，更为重要的问题是安全策略以及组信任关系的确定。因为 P2P 系统没有集中管理机制，所以系统中一个很重要的问题就是信任问题。虽然近期在信任关系管理方面已经作了大量的研究[45~47]，但是信任协商和安全策略还存在大量的问题值得研究。所以在 P2P 网络环境下，安全组通信密钥管理与组成员信任关系的研究不仅是必要的，而且是可行的。

无线通信技术的发展，使无线网络成为当前研究的一个热点问题。无线网络具有与生俱来的组播通信特性[48]，所以安全组播通信的研究可以应用在无线网络环境。但无线网络环境下的安全组通信密钥管理的研究要考虑到无线网络环境较之有线网络所具有的不同特点。无线网络相对有线网络的不同特点主要是：网络带宽、网络传输介质的可靠性。所以，无线环境下的安全组通信密钥管理的研究应注意考虑提高可靠性和降低网络带宽开销的问题。移动网络组用户节点的移动特性(mobility)增加了组通信密钥管理机制研究的难度。具有良好适应性的移动组通信中的密钥管理方式应该能够在不干扰网络无缝连接特性的情况下，在网络间移动，同时不会离开安全通信组。当用户从一个网络移动到另一个网络时，网络的可信任性是安全组通信要考虑的关键问题。

4.2.7.3　身份认证的研究

在安全组通信中，组通信数据源和通信组成员身份的认证是重要的安全特性，包括用户身份认证和数据源认证。通过通信组成员的身份验证，确保只有具有合法身份的用户可以加入通信组，参与通信组内的数据传输，确保非组成员无法生成有效的认证信息，进而无法冒充组成员接收组通信报文，可以拒非法企图窃取组通信数据者于通信组之外。通过数据源身份认证可以确保其身份的合法性，确保数据源对发送的数据负责，保证不可否认性。目前，最具代表性的身份验证方法是基于权威机构颁发的公钥证书的 PKI 认证方式。其他简单的认证方式也有，如基于预先共享密钥(pre-shared key)的认证方式等。根据应用安全性的要求，针对具体算法进行身份认证方面的研究，提高组通信的安全性具有重要的研究价值。如何将用户身份认证与安全组通信密钥管理机制相结合具有很大的研究空间。

4.2.7.4　安全多媒体传输中的密钥管理

组通信密钥管理方案可以结合视频安全组通信应用的具体特点进行相应的研究，以确

保视频组通信中的安全性、可靠性、健壮性和高效性，如针对小范围内的视频组通信应用（如视频会议），进行基于 Diffie-Hellman 算法的密钥协商方案的研究；针对组成员完全分布的视频组通信应用（如 pay-per-view 付费视频点播），进行安全组通信框架方案的研究。

现有组密钥管理的研究成果主要集中在系统的安全性的保证和实现上。在现有研究成果的基础上，应提高视频安全组通信系统方案的可靠性和健壮性。上述两种方案均非集中式控制，可以有效地避免由于组控制中心节点（GC）负载过重、故障等问题使之成为系统的瓶颈，降低系统的脆弱性；同时，引入延时重发和消息握手机制（密钥更新消息发送后，发送者等待接收者反馈确认信息，特定时间间隔内未收到确认信息，视情况重发密钥更新消息或将无反馈信息的节点作为失效节点来处理），在密钥分发、更新等过程中确保 re-key 消息可靠传递。

采用嵌入媒体信道传输方式（media-dependent channel），将密钥生成、分发、更新的消息包嵌入视频媒体数据流进行传输，减小密钥相关消息数据被嗅探、截获的可能性，提高系统的安全性，如图 4.2.16 所示。结合密钥消息嵌入，改进 re-key 消息格式，提高消息传递和密钥更新效率。

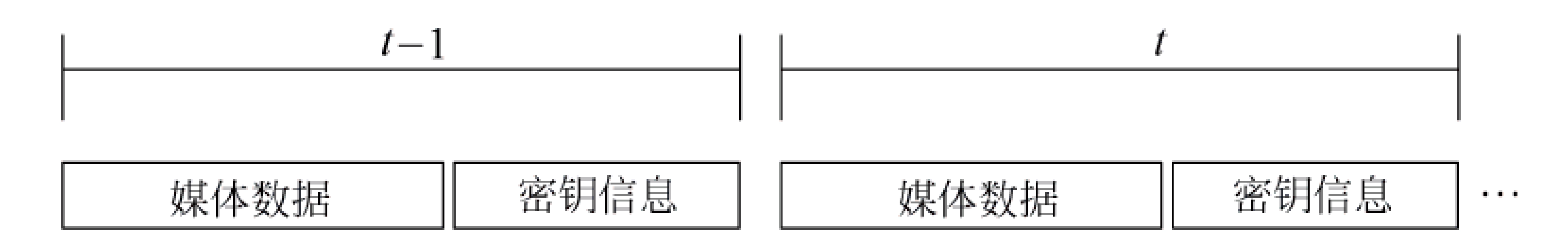


图 4.2.16　密钥消息嵌入媒体信道传输

4.2.7.5　特性间的权衡研究

通过特性间权衡的研究可以使密钥管理方案在多方面特性之间加以权衡，最终找到一种方案满足特定应用的需求。

权衡模式：在组通信密钥管理中，需要在管理效率和计算开销、管理效率和安全性、管理效率和组通信系统可靠性（密钥恢复）、存储开销和网络带宽占用等诸多因素之间进行权衡。如集中式密钥管理中，因为要保证用户离开时的前向隐私性，才采用层次（LKH）进行密钥的管理组织，而这种方法是牺牲密钥管理的效率换取安全性；而单向函数树（C-OFT）、单向函数链树（OFCT）是通过增加计算开销，换取密钥更新消息长度的减小。文献[2]采用的方法以提高通信开销换取控制器存储空间占用的减少。该方式提供一个可变的工作点，可以根据给出的要求进行设置。文献[4]提出的方法同样是以提高通信开销换取控制器存储空间占用的减少。还有的方法是放松系统抗同谋破解的力度换取存储开销和通信开销的减少。HySOR[5]采用混合结构，将 LKH 层次管理与 LORE 结合使用。LORE 具有良好的通信性能，却不能抗同谋破解而 LKH 具有抗同谋能力。两者使用的结合是在性能和抗同谋能力方面的权衡。

批处理模式：文献[49,50]提出在通信组中，以定时的批处理密钥更新模式替代每次组用户关系变化就进行一次密钥更新的操作模式，这实际上是安全性和性能之间的权衡。采用定时或者批处理密钥更新可以同时减小密钥服务器的处理开销和通信开销，同时提高密钥管理协议的性能和可扩展性，而在一定程度上损失了安全性。文献[49]提出的平面模式使用布尔最小化技术 BFM（Boolean function minimization）可以将多个用户离开需要的密

钥更新消息减少到 1 个。Kronos[31]是采用定期密钥更新的分散式密钥管理方式。组密钥在一定时间周期内产生，在当前周期内，所有组成员的变化均进行收集，在周期将结束、新组密钥分时进行处理。

自恢复模式：文献[3,9]提出了在不可靠的网络上进行大规模动态组播组密钥管理的方法。在不可靠的网络上，密钥更新消息很可能丢失而不能到达目的节点，在这种情况下，用户若请求组控制器进行重传，势必增加网络开销。Keystone[9]和 ELK[3]通过采用非交互的方式研究解决这一问题。用户无需与组控制器进行交互以获得丢失的密钥信息，也无需进行附加的重传操作，即可自身恢复密钥。它们通过在密钥分发后扩展附加密钥更新信息来防止丢包。Keystone 采用错误更正编码(error correction code)来生成以往组密钥信息，而 ELK 采用线索来生成更新后的密钥。同时，必须确保新用户不能通过此种方式恢复出以往的组密钥而破坏组播的后向隐私性。此外，两种模式是基于状态管理的。文献[10]提出了一种自恢复密钥分发机制，扩展了文献[11，12]中提出的子集差异密钥更新技术(subset difference re-keying techniques)。文献[10]提出的机制可以实现用户在离线一段时间的情况下，可在它重新返回时立刻恢复新的组播组密钥。

综上所述，对密钥管理方案的多方面特性之间进行权衡研究，可便于我们找到更加适合应用要求的密钥管理方式。

4.2.8　不同方案的应用环境

综上所述，在安全组通信应用中，我们应该根据应用特点选择、完善密钥管理方案：

(1) 集中式密钥管理方案便于管理和实现，密钥管理服务器和数据服务器可以集中、统一部署。但采用该类方案要注意解决中央服务器的单失效节点问题。

(2) 在用户节点运算处理能力较强，而网络带宽资源紧张的组通信应用系统中，应考虑采用以计算开销换取网络传输效率的密钥管理方案，如 C-OFT[18]，OFCT[23]等。

(3) 在不需要保证前向隐私性的组通信应用系统中，应优先考虑 GCP[21,22]，SMKD[27]等密钥管理方案，可以有效提高密钥管理的效率；在对抗同谋破解要求较低的情况下，可以考虑采用 C-FT[49]，D-FT[19]管理方案。

(4) 在多级数据服务器构建的组通信系统中，可以选择分散式密钥管理方式以获得不同的管理特性，见表 4.2.4。

表 4.2.4　特定需求方案总结

安全组通信应用需求	实 现 方 法	采 用 方 案
数据对非完全信任的中间服务节点保密	双重或者多重加密	DEP，CS 等
对密钥管理的效率要求高而对前向、后向隐私性要求较低	定期、批处理密钥更新	Kronos，MARKS 等
采用媒体相关隧道进行密钥传输	引入密钥恢复机制或应用层可靠性保证	Keystone，ELK 等
高鲁棒性，避免 1-affect-n 问题	分散控制、局部密钥更新	IOLUS 等

(5) 在要求平等参与组密钥 GK 生成的组通信系统中，应考虑采用分布式密钥协商方案。但该类方案一般需要较大的计算开销，需要用户节点具有较强的运算能力。

目前，组通信密钥管理研究多限于方案本身的安全性、可扩展性方面的研究，存在着与具体应用结合研究得不够等问题。在今后的研究中，把组通信密钥管理的研究与组通信的具体应用相结合，如视频安全点播、P2P 应用、无线传感器网络、移动网络等，以寻求最佳的密钥管理方案，仍具有重要的研究意义和实用价值。

此外，就组密钥管理方案本身而言，根据当前组密钥管理研究的热点问题和研究趋势来看，我们认为对如下 3 个主题展开深入的研究仍很有必要性：

(1) 组密钥分发协议的鲁棒性：如何在不可靠、易受攻击和开放的组通信环境中保证组密钥能被可靠地分发和更新，也即要求协议具备一定的容侵和容错特性。

(2) 组密钥分发协议的可扩展性：如何对较大规模的动态组用户群进行有效的组密钥管理。

(3) 组密钥分发协议的动态性能：如何在组用户频繁参与或退出组通信的情况下保持协议的动态稳定性。

围绕如上 3 个主题的研究，最近几年已有很大进展。但考虑到组通信的实际复杂性，组密钥管理的研究依然有很大的改善空间。组密钥分发协议的鲁棒性、可扩展性和动态性能仍然是组密钥管理研究工作中亟待解决的几个重要问题，特别是在复杂异构、开放、不可靠和易受攻击的实际组通信应用背景环境中。

4.3 基本的组密钥分发协议

通信加密密钥 TEK 通常是用一个公共密钥加密数据来实现的，TEK 被所有合法的组用户所共享。如果广播通道是开放的，网络很容易遭到非授权的访问。因此，即使数据被广播到整个网络，要求满足组播的机密性以防止非法用户有权访问机密的内容，而只有合法用户才能够解密数据。为了达到这一目的，源用户用对称密钥 TEK 加密数据，目的用户用 TEK 解密数据。此外，考虑到因用户的加入和离开使得用户拓扑结构发生动态变化，必须更新 TEK 以防止离开的用户访问未来的通信和新加入的用户访问以前的通信。组通信控制器 GC 将作为中央控制节点全权负责组通信密钥的创建、分发和组成员关系发生变化时的密钥更新。

组密钥分发协议的设计应该同时考虑协议的安全性和性能。就性能需求而言，协议设计应该考虑的重要性能因素包括：密钥更新消息的带宽消耗，组密钥更新消息长度，组用户存储的密钥数量，以及组用户的计算开销等。即协议应该具备较小的存储、计算和通信开销。基本的安全性需求则包括[51]：①组密钥的机密性(group key secrecy)：不是组中的活动用户无权获得用来解密组中广播数据的密钥；②前向隐私性(forward secrecy)：离开组的用户无权访问后续的组密钥，这样就保证了离开的用户不能解密后续的通信数据；③后向隐私性(backward secrecy)：一个新加入的用户进入会话后不能访问任何先前的组密钥，这样就保证了用户不能解密在其加入组之前发送的通信数据；④抗同谋破解(collusion freedom)：任何由欺骗用户组成的集合不能退出当前的活动 TEK。

在4.1节中,我们提及在本章中将重点研究的3个主题:组密钥分发协议的鲁棒性、可扩展性和动态性能。在本节中,我们将重点研究组密钥分发协议的动态性能,即在组用户频繁参与或退出组通信的情况下如何保持协议的动态稳定性。其他两个主题:协议的鲁棒性和可扩展性将在后续章节中逐步探讨。

如何解决用户拓扑频繁变化的密钥分发协议的动态稳定性问题?近年来,国际上的学者已经提出一些有效的方法,如LKH[15,17],OFT[18,23],MARKS[52],ELK[3]和SD(subset difference)[53,54],早期综述性的研究在文献[7,26]中可以看到,最近的是文献[51]。考虑到密钥更新消息的相互依赖,可撤销的组密钥分发机制被分为两类:无状态的和有状态的机制。在有状态机制中,一个有效用户的状态在当前密钥更新中将会影响它解密未来组密钥的能力。然而,在无状态机制中组密钥的更新只依赖于当前的密钥更新消息和用户的初始配置。一个不可撤销的用户即使只离线一会儿,也可以从以前的密钥更新消息中独立地解密出新的TEK,而无需依靠GC。在一些用户不是总在线或突然遭到包丢失的情况下,这种特点使得无状态机制非常有用。

Wallner等人[15]和Wong等人[17]提出了LKH (logical key hierarchy)机制,它是一种有效状态组用户的撤销机制,要求每一个用户存储$O(\log n)$个密钥,对每次的密钥更新操作GC只广播$2\log n$个消息,其中n是合法用户的数目。

无状态用户撤销机制首先由Fiat和Naor[5]提出,这种机制要求有$O(tn^2\log t)$个存储密钥和$O(t^2 n\log^2 t)$个消息,并且允许GC可以撤销任意一个用户,在这些用户中至多有t个能够联合起来得到TEK。Blundo等人[55,56]对广播加密提出了一种无条件的安全模型,并得到了计算和存储开销的最下界和最上界。后来,Naor等人[11]提出了两种无状态撤销机制,即CS和SD。如果N个用户有$\log N$个密钥,则CS机制可以撤销任意$O(R\log(N/R))$个消息中的R个用户。SD机制尽管将密钥更新消息数减少到$O(R)$,然而$O(\log N)$个加密操作却将用户存储开销增加到$O(\log^2 N)$。LSD (layered subset difference)不同于SD机制,它将存储开销从$O(\log^2 N)$减少到$O(\log^{1+\epsilon} N)$,然而却增加了通信开销。Wang等人提出了一种无状态机制[54],它用通信和计算开销减少了用户的存储要求。

针对开放的组通信环境中这些性能和安全性需求,提出了一种基本的组密钥分发协议B-GKDS(basic group key distribution scheme)。该协议的基本思想来源于Liu和Ning所提出的组密钥分发协议[57]中的个人秘密分发协议(personel secret distribution protocol)。我们在此基础上作了改进,使其能够适应于组通信中的密钥分发。尽管B-GKDS协议只是本文后续相关组密钥分发协议的原型,然而它提供了一种简捷、实用的组用户动态管理机制,显式(explicit)用户撤销机制。该机制能在组用户频繁参与或退出组通信的情况下保证协议良好的动态稳定性,并且能够满足基本的组通信安全需求,如组密钥的机密性、前向隐私性、后向隐私性和抗同谋破解。

B-GKDS协议中的组用户撤销算法具有较小的计算和通信开销,这使得B-GKDS协议能够适应于网络中动态用户拓扑结构多变的组通信环境。在B-GKDS协议中,组密钥被周期性地更新而不是在每个用户的拓扑变化中更新。周期性或批量的组密钥更新能够显著地减少GC和用户之间的计算和通信开销,因此能够有效地改善组密钥分发协议的性能和扩展性。

4.3.1 信息论概述

我们将利用信息熵概念来定义和描述组密钥分发协议模型,并利用信息泄露的分析技术来讨论相关协议的安全性。因此,我们将在本节对离散熵和互信息的基本概念及其主要性质进行简要介绍,主要内容来源于文献[59]中的相关章节。

信息论的创始人 Shannon 在其 1948 年发表的信息论奠基性论文《通信的数学理论》中提出了两个重要的概念:熵(entropy)和互信息(mutual information)。利用这两个概念,Shannon 对通信系统进行了理论分析,取得了通信技术史上划时代的重要成果。

4.3.1.1 离散熵定义

熵在信息论中是一个非常重要的概念,它是不确定性的一种度量。

定义 4.3.1 假定 $\boldsymbol{X}$ 为一定义在集合 X 上的离散随机变量;若 $\boldsymbol{X}$ 中各事件的随机概率分布为 $\{P(x)\}_{x\in X}=\{p_1,p_2,\cdots,p_{|X|}\}$,且满足:

$$\sum_{x\in X} P_{\boldsymbol{X}}(x)=1,\quad 0\leqslant P_{\boldsymbol{X}}(x)\leqslant 1,$$

则随机变量 $\boldsymbol{X}$ 的离散熵被定义为

$$H(\boldsymbol{X})=H(p_1,p_2,\cdots,p_{|X|})=-\sum_{i=1}^{|X|} p_i\log p_i \tag{4.3.1}$$

依据定义,我们知道熵 $H(\boldsymbol{X})$ 可以作为信息的度量。当有多个随机变量时,如 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$,为区别不同随机变量的熵,可将熵写成 $H(\boldsymbol{X})$ 和 $H(\boldsymbol{Y})$,以分别表示随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的熵。

熵 $H(\boldsymbol{X})$ 可以看作是 $|X|$ 维概率矢量 $p=(p_1,p_2,\cdots,p_{|X|})$ 的函数,称为熵函数。

性质 4.3.1 熵函数 $H(\boldsymbol{X})$ 具有以下重要性质:

(1) 对称性:概率矢量 $p=(p_1,p_2,\cdots,p_{|X|})$ 各分量 $p_1,p_2,\cdots,p_{|X|}$ 的次序任意改变时,熵函数 $H(\boldsymbol{X})$ 的值不变,即熵值只与集合 X 总体上的统计特征有关;

(2) 非负性:熵函数是一个非负量,即:$H(\boldsymbol{X})=H(p_1,p_2,\cdots,p_{|X|})\geqslant 0$;

(3) 确定性:集合 X 中只要有一个必然事件,其熵值必为 0;

(4) 极值性:当集合 X 中各事件以等概率出现时,其熵值 $H(\boldsymbol{X})$ 为最大,即有:

$H(\boldsymbol{X})=H(p_1,p_2,\cdots,p_{|X|})\leqslant H(1/|X|,1/|X|,\cdots,1/|X|)=\log|X|$。

从概率论的角度来看,某一特征的熵值越小,则包含的确定性信息越多。

4.3.1.2 联合熵与条件熵

一个随机变量的不确定性可以用熵来表示。利用多元随机变量的联合概率分布和条件分布,则可以得到联合熵与条件熵。

定义 4.3.2 设二元随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 分别为定义在集合 X 和集合 Y 上的离散随机变量,若 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的联合概率分布为 $\{P_{\boldsymbol{XY}}(x,y)\}_{x\in X,y\in Y}$,且满足:

$$\sum_{x\in X,y\in Y} P_{\boldsymbol{XY}}(x,y)=1,\quad 0\leqslant P_{\boldsymbol{XY}}(x,y)\leqslant 1 \tag{4.3.2}$$

则二元随机变量 $(\boldsymbol{X},\boldsymbol{Y})$ 的联合熵 $H(\boldsymbol{XY})$ 定义为

$$H(\boldsymbol{XY})=-\sum_{x\in X}\sum_{y\in Y} P_{\boldsymbol{XY}}(x,y)\log P_{\boldsymbol{XY}}(x,y) \tag{4.3.3}$$

对随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 而言，联合熵能对二元随机变量的不确定性进行度量。

定义 4.3.3　二元随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的条件熵 $H(\boldsymbol{X}|\boldsymbol{Y})$ 被定义为

$$H(\boldsymbol{X} \mid \boldsymbol{Y}) = -\sum_{y \in Y} \sum_{x \in X} P_{\boldsymbol{Y}}(y) P_{\boldsymbol{X}|\boldsymbol{Y}}(x \mid y) \log P_{\boldsymbol{X}|\boldsymbol{Y}}(x \mid y) \tag{4.3.4}$$

条件熵 $H(\boldsymbol{X}|\boldsymbol{Y})$ 表示在已知一个随机变量的情况下，对另一个随机变量的不确定性的度量。

性质 4.3.2　联合熵 $H(\boldsymbol{XY})$ 和条件熵 $H(\boldsymbol{X}|\boldsymbol{Y})$ 及 $H(\boldsymbol{Y}|\boldsymbol{X})$ 之间满足关系：

$$H(\boldsymbol{XY}) = H(\boldsymbol{Y}) + H(\boldsymbol{X} \mid \boldsymbol{Y}) \tag{4.3.5}$$

$$H(\boldsymbol{XY}) = H(\boldsymbol{X}) + H(\boldsymbol{Y} \mid \boldsymbol{X}) \tag{4.3.6}$$

性质 4.3.3　二元随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 相互独立时，联合熵 $H(\boldsymbol{XY})$、条件熵 $H(\boldsymbol{X}|\boldsymbol{Y})$ 和 $H(\boldsymbol{Y}|\boldsymbol{X})$ 满足关系：

$$H(\boldsymbol{X} \mid \boldsymbol{Y}) = H(\boldsymbol{X}) \tag{4.3.7}$$

$$H(\boldsymbol{Y} \mid \boldsymbol{X}) = H(\boldsymbol{Y}) \tag{4.3.8}$$

$$H(\boldsymbol{XY}) = H(\boldsymbol{X}) + H(\boldsymbol{Y}) \tag{4.3.9}$$

这表明，当随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 相互独立时，其联合熵等于单个随机变量的熵之和，而条件熵则等于无条件熵。

性质 4.3.4　一般情况下，联合熵、无条件熵、条件熵之间存在如下不等关系：

$$H(\boldsymbol{X} \mid \boldsymbol{Y}) \leqslant H(\boldsymbol{X}) \tag{4.3.10}$$

$$H(\boldsymbol{Y} \mid \boldsymbol{X}) \leqslant H(\boldsymbol{Y}) \tag{4.3.11}$$

$$H(\boldsymbol{XY}) \leqslant H(\boldsymbol{X}) + H(\boldsymbol{Y}) \tag{4.3.12}$$

这表明，条件熵在一般情况下总是小于无条件熵。直观上看，由于事物之间总是有联系的；因此一般而言，对随机变量 $\boldsymbol{X}$ 的了解总是能使随机变量 $\boldsymbol{Y}$ 不确定性减少。同样，对随机变量 $\boldsymbol{Y}$ 的了解也会减少 $\boldsymbol{X}$ 的不定性。

性质 4.3.5　若随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间存在确定的函数关系，且 $\boldsymbol{X}$ 可以完全确定 $\boldsymbol{Y}$，则有：

$$H(\boldsymbol{Y} \mid \boldsymbol{X}) = 0 \tag{4.3.13}$$

$$H(\boldsymbol{XY}) = H(\boldsymbol{X}) \tag{4.3.14}$$

若 $\boldsymbol{Y}$ 可以完全确定 $\boldsymbol{X}$，则有：

$$H(\boldsymbol{X} \mid \boldsymbol{Y}) = 0 \tag{4.3.15}$$

$$H(\boldsymbol{XY}) = H(\boldsymbol{Y}) \tag{4.3.16}$$

4.3.1.3　互信息

对两个随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$，它们之间存在某种统计依赖关系。未知 $\boldsymbol{Y}$ 时，$\boldsymbol{X}$ 的不确定度为 $H(\boldsymbol{X})$；已知 $\boldsymbol{X}$ 时，$\boldsymbol{Y}$ 的不确定度为 $H(\boldsymbol{X}|\boldsymbol{Y})$，且有 $H(\boldsymbol{X}|\boldsymbol{Y}) \leqslant H(\boldsymbol{X})$。因此，在了解 $\boldsymbol{Y}$ 后，$\boldsymbol{X}$ 的不确定度减少为 $H(\boldsymbol{X}) - H(\boldsymbol{X}|\boldsymbol{Y})$。

定义 4.3.4　为确切地描述离散随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间相互提供的信息量，将变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间的互信息 $I(\boldsymbol{X};\boldsymbol{Y})$ 或 $I(\boldsymbol{Y};\boldsymbol{X})$ 分别定义为

$$I(\boldsymbol{X};\boldsymbol{Y}) = H(\boldsymbol{X}) - H(\boldsymbol{X} \mid \boldsymbol{Y}) \tag{4.3.17}$$

$$I(\boldsymbol{Y};\boldsymbol{X}) = H(\boldsymbol{Y}) - H(\boldsymbol{Y} \mid \boldsymbol{X}) \tag{4.3.18}$$

性质 4.3.6 随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间互信息满足如下关系：

$$I(\boldsymbol{Y};\boldsymbol{X}) = I(\boldsymbol{X};\boldsymbol{Y}) \tag{4.3.19}$$

$$0 \leqslant I(\boldsymbol{X};\boldsymbol{Y}) \leqslant \min(H(\boldsymbol{X}),H(\boldsymbol{Y})) \tag{4.3.20}$$

性质 4.3.7 当随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间相互统计独立时，有：

$$I(\boldsymbol{X};\boldsymbol{Y}) = I(\boldsymbol{Y};\boldsymbol{X}) = 0 \tag{4.3.21}$$

性质 4.3.8 若随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间存在确定的函数关系，则有：

(1) 若 $\boldsymbol{X}$ 可以完全确定 $\boldsymbol{Y}$，此时 $H(\boldsymbol{Y}|\boldsymbol{X})=0$，则 $I(\boldsymbol{X};\boldsymbol{Y})=H(\boldsymbol{Y})$；

(2) 若 $\boldsymbol{Y}$ 可以完全确定 $\boldsymbol{X}$，此时 $H(\boldsymbol{X};\boldsymbol{Y})=0$，则 $I(\boldsymbol{X};\boldsymbol{Y})=H(\boldsymbol{X})$。

因此，互信息 $I(\boldsymbol{X};\boldsymbol{Y})$是对随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间统计依赖程度的度量，这也是互信息的另外一层含义。

4.3.2 基本的组密钥分发协议

基本的组密钥分发协议属于集中式的安全组密钥管理方案。在该协议中，由单一通信实体担当中央控制节点（组控制器 GC）来负责组密钥的创建、分发和组成员关系发生变化时的组密钥更新。

本节假定组通信系统的组成员和 GC 之间有一个统一的全局时钟。令 m 是组通信系统的活动周期。为方便讨论，我们假设 m 是一个整数，系统在 0 时刻启动，在 m 时刻停止。相应地，组通信系统的活动周期将被划分成 m 次会话。我们认为，组通信时间周期的限制从相当程度上是合理的，它适用于在不同时间段采用不同密钥加密的安全组通信应用中，如付费电视节目在不同时间段采用不同的组密钥加密。

我们首先利用信息熵形式化地定义和描述了基本的组密钥分发模型 $\boldsymbol{D}_B(\boldsymbol{U},t,m)$；而后我们将依次讨论 B-GKDS 协议的基本体系结构、活动组用户的初始化、组密钥的更新和恢复机制以及组用户的动态参与特性。

4.3.2.1 B-GKDS 协议的信息熵模型

假设 $\boldsymbol{U}$ 是所有可能的组通信用户的集合，每个用户 $U_i \subseteq \boldsymbol{U}$ 在加入第 j 次会话之前将从 GC 获得一个秘密私钥 S_i，该私钥将在用户 U_i 被 GC 撤销前一直为通信系统所使用。设 $R_j \subseteq \boldsymbol{U}$ 是第 j 次会话中由 GC 所撤销的组用户，$J_j \subseteq \boldsymbol{U}$ 是第 j 次会话中新加入的组用户，$G_j \subseteq \boldsymbol{U}$ 是第 j 次会话中合法的组成员，则有

$$G_j = (G_{j-1} \cup J_j) \backslash R_j \tag{4.3.22}$$

在每次组会话过程中，GC 将通过广播信道传输密钥更新消息 B_j。对每个组用户 U_i 而言，组密钥 K_j 完全可由 B_j 及其秘密私钥 S_i 决定。组密钥的更新要求协议提供一种安全的方法来传输广播通道上的密钥更新消息，且满足任意用户 $U_i \in G_i$ 能够解密该消息，而任何在 R_j 中的用户（$|R_j| \leqslant t$）即使他们以任意模式联合也不能解密该消息[60~62]。

我们假定 B-GKDS 协议的所有操作均在有限域 F_q 中进行，其中 q 是一个远大于 m 的大素数。因此用户的秘密私钥 $S_i \subseteq F_q$ 是 F_q 的子集；组密钥 $K_j \in F_q$ 则是 F_q 集中的元素。设随机变量 $\boldsymbol{S}_i$，$\boldsymbol{B}_j$ 和 $\boldsymbol{K}_j$ 分别表示用户 U_i 的个人私钥、GC 的广播更新消息以及第 j 次组会话密钥，则利用信息熵的概念，我们可以将安全的组密钥分发协议 B-GKDS 形式化地定义

如下。

定义 4.3.5　假定 $\boldsymbol{U}=\{U_1,U_2,\cdots,U_n\}$是所有可能的组用户集，$R_j\subseteq\boldsymbol{U}$ 是第 j 次会话中由 GC 所撤销的组用户，$J_j\subseteq\boldsymbol{U}$ 是第 j 次会话中新加入的组用户，$G_j=(G_{j-1}\cup J_j)\backslash R_j$ 是第 j 次会话中合法的组成员；m 表示组通信系统的最大会话次数，t 是 GC 所能撤销的最大用户数。则 $\boldsymbol{D}_B(\boldsymbol{U},t,m)$是一个基本的组密钥分发模型，若满足：

（1）对组用户 $U_i\in G_i$ 而言，组密钥 K_j 完全可由 B_j 及其私钥 S_i 决定，即：

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j,\boldsymbol{S}_i)=0 \tag{4.3.23}$$

（2）对任意用户子集 $B\subseteq\boldsymbol{U}$，$|B|\leqslant t$，集合 B 中的用户不可能获得用户 $U_k\notin B$ 的用户私钥 S_k，即：

$$H(\boldsymbol{K}_j,\boldsymbol{S}_k \mid \boldsymbol{B}_j,\boldsymbol{B}_{j-1},\cdots,\boldsymbol{B}_1,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{B}})=H(\boldsymbol{K}_j,\boldsymbol{S}_k) \tag{4.3.24}$$

（3）不可能单独从 GC 广播的密钥更新信息或组用户私钥，获得组密钥，即：

$$H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m \mid \boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_m)=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m) \tag{4.3.25}$$

$$H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m \mid \boldsymbol{S}_1,\boldsymbol{S}_2,\cdots,\boldsymbol{S}_n)=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m) \tag{4.3.26}$$

（4）$\boldsymbol{D}_B(\boldsymbol{U},t,m)$能同时安全地撤销最多 t 个用户：设每次在会话 j 被撤销的组用户集是 $R=R_j\cup R_{j-1}\cup\cdots\cup R_1(|R|\leqslant t)$，则 R 中的组用户不可能利用 GC 广播的组密钥更新信息 B_j 去恢复出当前的组通信密钥 K_j，即：

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j,\boldsymbol{B}_{j-1},\cdots,\boldsymbol{B}_1,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{R}})=H(\boldsymbol{K}_j) \tag{4.3.27}$$

定义 4.3.5 中的性质 1～性质 3 描述了 $\boldsymbol{D}_B(\boldsymbol{U},t,m)$作为一个组密钥分发模型所应满足的基本安全属性。定义中的性质 4 则形式化地描述了 $\boldsymbol{D}_B(\boldsymbol{U},t,m)$的用户撤销机制。考虑到组密钥分发协议的安全性需求，$\boldsymbol{D}_B(\boldsymbol{U},t,m)$还应满足组密钥的前向隐私性和后向隐私性。

定义 4.3.6　组密钥管理协议 $\boldsymbol{D}_B(\boldsymbol{U},t,m)$的前向隐私性和后向隐私性定义为：

（1）前向隐私性。假定 $B\subseteq R_r\cup R_{r-1}\cup\cdots\cup R_1$ 表示在会话 r 前撤销的组用户集，设 $|B|\leqslant t$，则 B 中任何被撤销的组用户之间的密谋均无法获得组密钥 $K_j(r\leqslant j\leqslant m)$，即：

$$\begin{aligned}&H(\boldsymbol{K}_r,\boldsymbol{K}_{r+1},\cdots,\boldsymbol{K}_m \mid \boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_m,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{R}},\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_{r-1})\\&=H(\boldsymbol{K}_r,\boldsymbol{K}_{r+1},\cdots,\boldsymbol{K}_m)\end{aligned} \tag{4.3.28}$$

（2）后向隐私性。假定 $F\subseteq U$ 为在会话 s 后加入组通信的活动用户集，设 $|F|\leqslant t$，则 F 中的任何组用户之间的密谋均无法获得组通信密钥 $K_j(1\leqslant j\leqslant s)$，即：

$$\begin{aligned}&H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_s \mid \boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_m,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{F}},\boldsymbol{K}_{s+1},\boldsymbol{K}_{s+2},\cdots,\boldsymbol{K}_m)\\&=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_s)\end{aligned} \tag{4.3.29}$$

4.3.2.2　B-GKDS 的体系结构

如图 4.3.1 所示，在 GC 和组用户之间，B-GBDS 协议包含两种基本的控制信息流：InitGroupKey 和 RefreshKey。InitGroupKey 是 GC 和单个组用户之间的单播消息，该消息在用户加入活动组的初始化阶段，用来分发对应的秘密参数给用户；而 RefreshKey 是 GC 在不同的组会话阶段产生的组密钥更新消息；它用来周期性地广播密钥更新消息给所有的活动组用户。

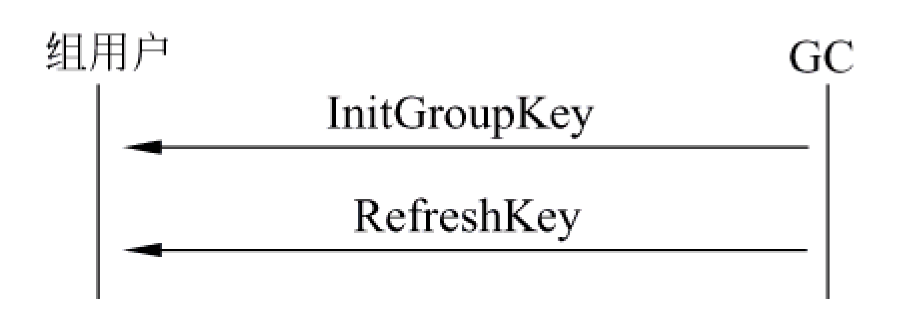


图 4.3.1　B-GKDS 协议中 GC 和组用户之间的控制信息流

4.3.2.3 B-GKDS组用户的初始化

在B-GKDS组用户的初始化阶段，组控制器GC随机地从有限域$F_q[x]$中选取m个度为t的屏蔽多项式：

$$\{h_j(x)=h_{0,j}+h_{1,j}x+\cdots+h_{t,j}x^t\}_{j=1,2,\cdots,m}\in F_q[x]。$$

随后，GC为每个在初始化阶段加入活动组的用户U_i产生m个秘密私钥$S_i=\{h_1(i),h_2(i),\cdots,h_m(i)\}$，并通过InitGroupKey消息将这$m$个秘密私钥$\{h_1(i),h_2(i),\cdots,h_m(i)\}$安全地分发给用户$U_i$。

$$\mathrm{GC}\rightarrow U_i:\{E_{\mathrm{MK}_i}(t\mid h_1(i)\mid\cdots\mid h_m(i)\mid)\mathrm{MAC}(t\mid h_1(i)\mid\cdots\mid h_m(i)\mid)\}$$

其中，t是时间戳；MAC(·)是产生消息验证码的散列函数(如MD5[63])，SHA-1[64]；MK_i是用户U_i与GC共享的主密钥，用于InitGroupKey消息的加密和验证。

U_i一旦接收到InitGroupKey消息，首先验证该消息的真伪；若为真，则U_i利用MK_i解密该消息并获得相应的秘密私钥$S_i=\{h_1(i),h_2(i),\cdots,h_m(i)\}$，并通过接收随后的密钥更新消息RefreshKey来同步地更新组会话密钥。

4.3.2.4 B-GKDS的组密钥更新机制

考虑组通信系统的第j次组密钥更新。设R_j表示在会话j阶段被撤销的组用户集，则$R=R_j\cup R_{j-1}\cup\cdots\cup R_1(|R|\leqslant t,R\cap G_j=\varnothing)$表示在会话$j$之前被撤销的所有组用户集。组密钥的更新机制要求提供一种安全机制来传输广播信道上的密钥更新消息RefreshKey，以满足任意的活动用户$U_i\in G_i$能够解密这个消息，而任何R中的非活动用户($|R|\leqslant t$)即使他们以任意模式密谋也不能解密该消息。为此，我们构造如下的广播消息B_j：

$$\mathrm{GC}\rightarrow *:\{w_j(x)\mid\{R\}\mid\mathrm{MAC}(\{R\}\mid w_j(x))\},$$

其中，多项式$w_j(x)=g_j(x)\cdot\mathrm{TEK}_j+h_j(x)$；$*$表示所有的活动组用户；$h_j(x)$是屏蔽多项式；$\mathrm{TEK}_j$是当前的组会话密钥；多项式$g_j(x)$则按如下方式构造：

$$g_j(x)=\prod_{r_i\in R}(x-r_i) \tag{4.3.30}$$

4.3.2.5 B-GKDS的组密钥恢复机制

当前的活动用户$U_i\in G_i$收到广播消息B_j时，首先利用广播消息B_j中的撤销用户集R，按式(4.3.30)构造多项式$g_j(x)$，然后分别计算多项式$w_j(x)$和$g_j(x)$在点i处的值$w_j(i)$和$g_j(i)$。由于$U_i\notin R$，故有$g_j(i)\neq 0$。因此，用户U_i可以利用在初始化阶段保留的秘密私钥$h_j(i)$以及$w_j(i)$和$g_j(i)$值进一步恢复出当前的组会话密钥TEK_j。

$$\mathrm{TEK}_j=(w_j(i)-h_j(i))/g_j(i) \tag{4.3.31}$$

而对于$R=R_j\cup R_{j-1}\cup\cdots\cup R_1(|R|\leqslant t)$中被GC撤销的用户$U_r$而言，由于$\{g_j(r)=0\mid\forall U_r\in R\}$，$w_j(r)=g_j(r)\cdot\mathrm{TEK}_j+h_j(r)=h_j(r)$，因此，即使他们密谋($|R|\leqslant t$)也难于破解该广播消息而恢复出当前的组会话密钥$\mathrm{TEK}_j$。

4.3.2.6 用户的动态参与机制

组用户的动态参与机制要求系统在用户加入或离开活动组的情况下，能够有效地保证

组会话密钥的安全性，即密钥的前向隐私性和后向隐私性。

用户离开：当用户离开活动组时，GC 需要将该用户注销。假设 $\boldsymbol{U}$ 是所有可能的组用户的集合，R 是被撤销的用户集，有 $R\subseteq\boldsymbol{U}$。组用户的撤销机制要求一种安全机制来传输组密钥更新消息，使得用户 $U_i\in\{U\backslash R\}$ 能够解密该消息，而任何在 R 中的用户即使他们以任意模式同谋也不能解密该消息。

B-GKDS 的组用户撤销机制隐含在组密钥的更新消息 RefreshKey 中。假设在第 j 次会话中，GC 需要撤销用户 U_r，则只需要将 U_r 包括在广播消息 B_j 的 $\{R\}$ 中，即 $U_r\in R$。由于 $\{g_j(r)=0\mid\forall U_r\in R\}$，用户 U_r 无法利用广播的密钥更新消息 B_j 和他自己先前保存的秘密 $h_j(r)$ 去恢复当前的组会话密钥 TEK_j。我们称这种主动的组用户撤销机制为显式用户撤销机制，因为被撤销的用户标识明确包含在组密钥的广播更新消息中。

用户加入：当用户 U_v 希望在会话阶段 j 加入活动组时，相应的处理机制如下：

(1) 用户 U_v 首先需要从 GC 获得加入活动组的许可。如果成功，GC 为 U_v 产生 $m-j+1$ 个秘密私钥 $\{h_j(v),h_{j+1}(v),\cdots,h_m(v)\}$，并通过 InitGroupKey 消息将秘密私钥 $\{h_j(v),h_{j+1}(v),\cdots,h_m(v)\}$ 安全地发送给用户 U_v。

$$\text{GC}\rightarrow U_v:\{E_{\text{MK}_v}(t\mid h_j(v)\mid\cdots\mid h_m(v)\mid)\text{MAC}(t\mid h_j(v)\mid\cdots\mid h_m(v)\mid)\},$$

其中，MK_v 是用户 U_v 与 GC 共享的主密钥，用于 InitGroupKey 消息的加密和验证。

(2) 用户 U_v 一旦接收到 InitGroupKey 消息，首先验证该消息的真伪；若为真，则 U_v 可以成功地加入到当前活动的组通信中，并通过接收随后的密钥更新消息 RefreshKey 来同步更新组会话密钥 TEK_j。

4.3.3　安全性和性能分析

4.3.3.1　安全性分析

这里我们分析 B-GKDS 能否满足组密钥的安全性要求。在如下定理的推导过程中，我们用随机变量 $\boldsymbol{K}_j(j=1,2,\cdots,m)$ 来表示对应的组会话密钥 $\text{TEK}_i(i=1,2,\cdots,m)$。

定理 4.3.1　组密钥管理协议 B-GKDS 能够同时安全地撤销最多 t 个用户；并且，就信息论范畴而言，B-GKDS 是一个无条件安全的组密钥分发协议。

证明：依据定义 4.3.5，假定 $\boldsymbol{U}=\{U_1,U_2,\cdots,U_n\}$ 是所有可能的组用户集；$R_j\subseteq\boldsymbol{U}$ 是第 j 次会话中由 GC 所撤销的组用户；$R=R_j\cup R_{j-1}\cup\cdots\cup R_1(|R|\leqslant t)$ 是在会话 j 被撤销的所有组用户集；$J_j\subseteq\boldsymbol{U}$ 是第 j 次会话中新加入的组用户；$G_j=(G_{j-1}\cup J_j)\backslash R_j$ 是第 j 次会话中合法的组成员；m 表示组通信系统的最大会话次数；t 是 GC 所能撤销的最大用户数。下面我们将证明 B-GKDS 协议能够满足 $\boldsymbol{D}_B(\boldsymbol{U},t,m)$ 所定义的 4 条基本性质。

首先我们证明性质 1。根据前述 B-GKDS 协议的组密钥恢复机制，我们可以知道，当活动用户 $U_i\in G_i$ 收到广播消息 B_j 时，它能利用广播消息 B_j 计算多项式 $w_j(x)$ 和 $g_j(x)$ 在点 i 处的值 $w_j(i)$ 和 $g_j(i)$。因为 $U_i\notin R$，有 $g_j(i)\neq0$ 成立；因此，用户 U_i 可以利用在初始化阶段保留的私钥 $h_j(i)$ 以及 $w_j(i)$ 和 $g_j(i)$ 值恢复出当前的组密钥 $\text{TEK}_j=(w_j(i)-h_j(i))/g_j(i)$。即对一个活动的组用户 $U_i\in G_i$ 而言，它完全可以利用当前的广播消息 B_j 及其私钥 S_i 恢复出组密钥 K_j，即有如下结论成立：

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j, \boldsymbol{S}_i) = 0。$$

其次，我们证明性质 2。对任意用户子集 $B \subseteq \boldsymbol{U}(|B| \leqslant t)$中的所有用户联合，最多能知道多项式 $h_j(x)$中的 t 个点。由于$\{h_j(x) \mid j=1,2,\cdots,m\}$是组控制器 GC 随机地从 $F_q[x]$中选取的度为 t 的屏蔽多项式，因此，集合 B 中所有用户不可能利用多项式的插值方法去重构 $h_j(x)$。即任何集合 $B \subseteq \boldsymbol{U}(|B| \leqslant t)$中的用户不可能通过密谋的方式获得组用户 $U_k \notin B$ 的秘密私钥 S_k，即：

$$H(\boldsymbol{K}_j, \boldsymbol{S}_k \mid \boldsymbol{B}_j, \boldsymbol{B}_{j-1}, \cdots, \boldsymbol{B}_1, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{B}}) = H(\boldsymbol{K}_j, \boldsymbol{S}_k)。$$

再次，我们证明性质 3。组密钥 $\mathrm{TEK}_j(1 \leqslant j \leqslant m)$是 GC 随机产生的，广播消息$\{\boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_m\}$中的多项式 $w_j(x) = g_j(x) \cdot \mathrm{TEK}_j + h_j(x)$并未泄露任何关于 $\mathrm{TEK}_j(1 \leqslant j \leqslant m)$的信息。因此，单独从 GC 广播的密钥更新信息是不可能获得组密钥的，即：

$$H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m \mid \boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_m) = H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m)。$$

另外，多项式$\{h_j(x) \mid j=1,2,\cdots,m\} \in F_q[x]$是 GC 随机地从 $F_q[x]$中产生的；组密钥 $\mathrm{TEK}_j(1 \leqslant j \leqslant m)$也是 GC 随机产生的，其完全独立于$\{h_j(x) \mid j=1,2,\cdots,m\}$，也即完全独立于组用户 U_i 的秘密 $S_i = \{h_1(i), h_2(i), \cdots, h_m(i)\}$。因此，单独从组用户的秘密私钥是不可能获得组密钥的，即：

$$H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m \mid \boldsymbol{S}_1, \boldsymbol{S}_2, \cdots, \boldsymbol{S}_n) = H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m)。$$

最后，我们证明性质 4，即 B-GKDS 协议能够同时安全地撤销最多 t 个用户。在考察 GC 广播第 j 次密钥更新消息 B_j 时，对于 $R = R_j \cup R_{j-1} \cup \cdots \cup R_1(|R| \leqslant t)$中被 GC 撤销的用户 U_r 而言，由于$\{g_j(r) = 0 \mid \forall U_r \in R\}$，因此有：

$$w_j(r) = g_j(r) \cdot \mathrm{TEK}_j + h_j(r) = h_j(r) \tag{4.3.32}$$

这导致用户 $U_r \in R$ 无法通过计算恢复出组密钥 TEK_j。

另一方面，若集合 $R(|R| \leqslant t)$中所有被撤销的用户进行密谋；对活动组的第 $j \in [1, m]$次会话，尽管所有这些用户通过密谋可以知道$\{h_j(r_i) \mid r_i \in R, j \in [1, m]\}$和 $w_j(x)$。然而，我们可以随机地选取 TEK'_j，并构造：

$$h'_j(x) = g_j(x) \cdot \mathrm{TEK}_j + h_j(x) - g_j(x) \cdot \mathrm{TEK}'_j \tag{4.3.33}$$

$h'_j(x)$是合理的，因为它能保证如下等式的成立：

$$w_j(x) = g_j(x) \cdot \mathrm{TEK}'_j + h'_j(x) \tag{4.3.34}$$

$$h'_j(r_i) = g_j(r_i) \cdot \mathrm{TEK}_j + h_j(r_i) - g_j(r_i) \cdot \mathrm{TEK}'_j = h_j(r_i) \tag{4.3.35}$$

这表明，任何一个随机产生的数 TEK'_j 都可能是 R 中所有被撤销用户密谋所得到的组密钥。因此，性质 4 成立，即

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j, \boldsymbol{B}_{j-1}, \cdots, \boldsymbol{B}_1, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{R}}) = H(\boldsymbol{K}_j)。$$

定理 4.3.2 组密钥管理协议 B-GKDS 能够满足前向隐私性和后向隐私性。

证明：依据定义 4.3.6，我们依次讨论 $\boldsymbol{D}_B(\boldsymbol{U}, t, m)$的前向和后向隐私性。

前向隐私性：假定 $B \subseteq R_r \cup R_{r-1} \cup \cdots \cup R_1(|B| \leqslant t)$表示在会话 r 前(包括 r)被 GC 撤销的组用户集。由于所有被撤销的用户 $U_r \in B$ 均包含在组密钥更新消息 RefreshKey 的$\{R\}$子项中；依据密钥更新消息中多项式 $g_j(x)$构造方式可知，等式$\{g_j(r) = 0 \mid \forall U_r \in R\}$和 $w_j(r) = h_j(r)$成立；进而，任何用户 $U_r \in R$ 均无法利用广播的密钥更新消息 B_j 及其先前保存的秘密 $h_j(r)$去恢复当前的组密钥 TEK_j，即下式关于组密钥 TEK_j 的计算对 $U_r \in R$ 的用户而言是不可行的，因为：①$g_j(r) = 0$；②$w_j(r) - h_j(r) = 0$。

$$\mathrm{TEK}_j = (w_j(r) - h_j(r))/g_j(r)。$$

考察 B 中所有用户的同谋：对 B 中所有用户 $U_b \in B$，由于 $|B| \leqslant t$，他们不可能利用这些私钥 $\{h_j(b) | r \leqslant j \leqslant m, U_b \in B\}$ 去构建度为 t 的多项式 $\{h_j(x) | r \leqslant j \leqslant m\}$。因此，只要 $|B| \leqslant t$，B 中组用户之间便不可能以同谋的方式获取组密钥 $\{\mathrm{TEK}_j | r \leqslant j \leqslant m\}$。即有：

$$H(\boldsymbol{K}_r, \boldsymbol{K}_{r+1}, \cdots, \boldsymbol{K}_m \mid \boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_m, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{R}}) = H(\boldsymbol{K}_r, \boldsymbol{K}_{r+1}, \cdots, \boldsymbol{K}_m)。$$

最后，考虑到 GC 所产生的组会话密钥 $\{K_j | 1 \leqslant j \leqslant m\}$ 的随机独立性，根据信息熵概念可以有 $H(\boldsymbol{K}_{s+1}, \boldsymbol{K}_{s+2}, \cdots, \boldsymbol{K}_m | \boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_s) = H(\boldsymbol{K}_{s+1}, \boldsymbol{K}_{s+2}, \cdots, \boldsymbol{K}_m)$ 成立。因此，B 中任何组用户之间的密谋也无法获得组密钥 $\{\mathrm{TEK}_j | r \leqslant j \leqslant m\}$，即如下结论成立：

$$H(\boldsymbol{K}_r, \boldsymbol{K}_{r+1}, \cdots, \boldsymbol{K}_m \mid \boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_m, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{R}}, \boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_{r-1}) = H(\boldsymbol{K}_r, \boldsymbol{K}_{r+1}, \cdots, \boldsymbol{K}_m)。$$

后向隐私性：假定 $F \subseteq \boldsymbol{U}$，$|F| \leqslant t$ 为在会话 s 后(包括 s)加入活动组的用户集。由组用户的动态加入规则可知：F 中任何在会话 $k(s \leqslant k \leqslant m)$ 加入活动组的用户 U_f 只能获得私钥 $S_f = \{h_j(f) | k \leqslant j \leqslant m\}$，而无法获得在会话 $k(s \leqslant k \leqslant m)$ 前的秘密私钥 $\{h_j(f) | 1 \leqslant j < k\}$，进而 U_f 不可能从广播更新消息 $\{B_j | 1 \leqslant j < s\}$ 恢复对应的组密钥 $K_j(1 \leqslant j \leqslant k)$。另外，对于 F 中所有用户的同谋，由于 $|F| \leqslant t$，他们是不可能利用这些私钥 $\{h_j(f) | s \leqslant j \leqslant m, U_f \in F\}$ 重建度为 t 的多项式 $\{h_j(x) | s \leqslant j \leqslant m\}$ 的；因此，只要 $|F| \leqslant t$，F 中组用户之间不可能以同谋方式获取组密钥 $\{\mathrm{TEK}_j | 1 \leqslant j < s\}$。此外，考虑到 GC 产生组会话密钥 $\{K_j | 1 \leqslant j \leqslant m\}$ 的随机独立性，有 $H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_s | \boldsymbol{K}_{s+1}, \boldsymbol{K}_{s+2}, \cdots, \boldsymbol{K}_m) = H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_s)$。因此，$F$ 中任何组用户之间的密谋也无法获得组通信密钥 $\{\mathrm{TEK}_j | 1 \leqslant j < s\}$，即：

$$H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_s \mid \boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_m, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{F}}, \boldsymbol{K}_{s+1}, \boldsymbol{K}_{s+2}, \cdots, \boldsymbol{K}_m) = H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_s)。$$

因此，B-GKDS 的动态用户参与算法提供了一种保证前向/后向隐私性的有效方法，也即，B-GKDS 的组密钥分发机制能够满足前向/后向隐私性需求。

4.3.3.2 性能分析

表 4.3.1 分别对 B-GKDS 协议的性能和 Stadden 的组密钥分发协议[10]进行了对比分析。

表 4.3.1　性能对比

	通信开销(组播)	通信开销(单播)	存储开销
B-GKDS	$O(t\log q)$	$O(m\log q)$	$O(m\log q)$
Stadden[10]	$O(t^2\log q)$	$O(m\log q)$	$O(m\log q)$

存储开销：在初始化阶段，每个组用户 U_i 需要存储自己的身份标识 i 和掩码多项式 $\{h_j(x)\}_{j=1,2,\cdots,z} \in F_q[x]$ 在点 i 处的值 $\{h_1(i), h_2(i), \cdots, h_z(i)\}$。因此，对每个用户 U_i 而言，其存储复杂度为 $O(m\log q)$，它和 Stadden 的组密钥分发协议的存储复杂度基本一样。

通信开销：广播消息 $\boldsymbol{B}_j$ 包括在第 j 次会话中由 GC 所撤销的组用户标识集和一个 t 维多项式 $w_j(x)$；因此，B-GKDS 的通信开销是 $O(t\log q)$，而 Stadden 的组密钥分发协议则为 $O(t^2\log q)$。显然，B-GKDS 协议使得 GC 和组用户之间的广播通信开销得到了显著优化，因为广播消息包的大小被减少到 $O(t\log q)$。

值得一提的是，同样的密钥分发机制，Liu 和 Ning 的协议[57]主要用于组用户个人秘密

的分发(personel secret distribution,PSD),因此,在表4.3.1中,我们没有将此协议列入性能的对比分析之中。实际上,PSD协议的广播通信开销近似为$2t\log q$,而B-GKDS协议的广播通信开销则近似为$t\log q$,性能还是获得了一定程度的优化。

4.4 自愈的组密钥分发协议

在基本的组密钥分发协议B-GKDS中,我们通过引入一种简洁而高效的无状态用户撤销管理机制,初步解决了组密钥分发协议的动态性问题。本节重点研究的主题是组密钥分发协议的鲁棒性,以期能够提供一种可靠的组密钥更新机制。事实上,除协议的动态稳定性主题研究以外,近年来一些研究工作也密切关注到组密钥的自愈性(self-healing)问题,即组用户自身能够从最近的密钥更新消息中恢复先前丢失的组会话密钥。这对于当前开放、异构、不可靠和易受攻击的组通信环境而言,有效的组密钥自愈算法更显重要。Stadden等人基于二维t次二项式首先提出一种自愈的组密钥分发机制[10],该方法后来被Liu和Ning所改进[57]。Blundo等人[52,53]则进一步指出了文献[57,58]的缺陷,并提出另一种组密钥的自愈机制。在文献[53,54,65]中,一些自愈方法被引入到基于子集差(subset difference)技术的组密钥更新协议中。

自愈的组密钥分发机制同样应该考虑安全性和性能。基本的安全性需求包括组密钥的机密性、前向隐私性、后向隐私性和抗同谋破解。此外,就性能而言,协议应该具备较小的存储、计算和通信开销。例如,低的通信带宽(无线链路)、通信链路较高的误码率和丢包率以及组用户的动态拓扑变化都会增加服务中断的概率,使组用户无法与GC正常通信,最终将导致协议失败。因此,协议要求:①较小的组密钥更新消息;②GC和用户之间尽量少的消息交互次数;③此外,组密钥的更新机制不能要求组用户存储大量的密钥和过于繁重的计算。

在基本组密钥分发协议B-GKDS的基础上,针对不可靠、易受攻击、开放的组通信环境,提出了一种自愈的组密钥分发协议S-GKDS(self-healing group key distribution scheme)。S-GKDS协议完全是B-GKDS协议的扩展,它完整地继承了B-GKDS协议的基本特性,如安全性和组用户动态参与机制的灵活性。协议的自愈机制是基于单向哈希链(one-way directional Hash chains,ODHC);ODHC的单向性提供了一种具有较小计算和通信开销的组密钥自愈算法。尽管密钥更新消息很可能在传输过程中丢失,但用户仍然可以通过单向函数和最近收到的组密钥更新消息来恢复先前丢失的组密钥更新消息。这种优良的特性使得S-GKDS协议对丢包率和错码率较高的组通信环境具有更好的自适应性。

在S-GKDS协议中,组密钥将被周期性地更新而不是在每个用户的拓扑变化时更新;周期性或批量地密钥更新能够显著地减少GC和用户之间的计算和通信开销,因此能够显著地改善组密钥分发协议的性能和扩展性。此外,为了保证组通信过程中数据的稳定传输,S-GKDS协议还引入了一种平滑的组密钥更新技术,使得在组密钥的动态切换过程中不会干扰正在传输的数据。

此外还讨论了S-GKDS协议在广播信道下的性能和安全性。性能分析结果表明,该协议能够有效地容忍通信信道中较高的丢包率和错码率。另外,在S-GKDS协议的信息熵模

型基础上，利用信息泄露的分析技术，证明了该协议在信息论范畴内是无条件安全的，即S-GKDS协议能够满足组通信的安全性需求，可以有效地保证组密钥的前向/后向隐私性。

因此，S-GKDS 协议具有较好的性能和安全性，它可被应用在不可靠的通信网络中，如无线网络、移动网络（NEMO 网）和无线传感器网络。

4.4.1　S-GKDS 协议的信息熵模型

与定义 4.3.5 相似，在此，我们给出 S-GKDS 协议的信息熵模型。S-GKDS 协议完全是 B-GKDS 协议的扩展，它在基本组密钥分发协议 B-GKDS 的基础上，引入了组密钥的自愈机制。因此，在如下 S-GKDS 协议的信息熵模型的定义中，我们主要扩展了对自愈的形式化定义，其他与基本的组密钥分发协议 B-GKDS 的信息熵模型一致。

同样地，假设随机变量 $\boldsymbol{S}_i$，$\boldsymbol{B}_j$ 和 $\boldsymbol{K}_j$ 分别表示组用户 U_i 的个人私钥、GC 的广播更新消息以及第 j 次组会话密钥，则利用信息熵的概念，我们可以将自愈的组密钥分发协议 S-GKDS模型 $\boldsymbol{D}_S(\boldsymbol{U},t,m)$形式化地定义如下：

定义 4.4.1　假定 $\boldsymbol{U}=\{U_1,U_2,\cdots,U_n\}$是所有可能的组通信用户的集合，$m$ 是组通信系统的最大会话次数，t 是 GC 所能主动撤销的最大用户数，则 $\boldsymbol{D}_S(\boldsymbol{U},t,m)$是一个自愈的组密钥分发模型，若如下条件能够满足：

（1）对组用户 $U_i\in G_i$ 而言，组密钥 K_j 完全可以由 B_j 及其私钥 S_i 所决定，即：

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j,\boldsymbol{S}_i)=0 \tag{4.4.1}$$

（2）对任意用户子集 $B\subseteq\boldsymbol{U}$，$|B|\leqslant t$，集合 B 中的用户不可能获得用户 $U_k\notin B$ 的用户私钥 S_k，即：

$$H(\boldsymbol{K}_j,\boldsymbol{S}_k \mid \boldsymbol{B}_j,\boldsymbol{B}_{j-1},\cdots,\boldsymbol{B}_1,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{B}})=H(\boldsymbol{K}_j,\boldsymbol{S}_k) \tag{4.4.2}$$

（3）不可能单独从 GC 广播的密钥更新信息或组用户私钥获得组密钥，即：

$$H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m \mid \boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_m)=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m) \tag{4.4.3}$$

$$H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m \mid \boldsymbol{S}_1,\boldsymbol{S}_2,\cdots,\boldsymbol{S}_n)=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m) \tag{4.4.4}$$

（4）$\boldsymbol{D}_S(\boldsymbol{U},t,m)$能够同时安全撤销最多 t 个用户的能力：设每次会话 j 被撤销的组用户集是 $R=R_j\cup R_{j-1}\cup\cdots\cup R_1$（$|R|\leqslant t$），则 R 中的组用户不可能利用 GC 广播的组密钥更新信息 B_j 去恢复出当前的组通信密钥 K_j，即：

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j,\boldsymbol{B}_{j-1},\cdots,\boldsymbol{B}_1,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{R}})=H(\boldsymbol{K}_j) \tag{4.4.5}$$

（5）对用户 $U_i\in G_r$，若其在会话 r 收到密钥更新消息$\{B_r|1\leqslant r<m\}$，但在会话 s 前一直没有被撤销（$r<s\leqslant m$），则该用户能够利用会话 s 收到的密钥更新消息$\{B_l|r\leqslant l\leqslant s\}$恢复所有的组通信密钥$\{K_l|r\leqslant l\leqslant s\}$，即：

$$H(\boldsymbol{K}_r,\boldsymbol{K}_{r+1},\cdots,\boldsymbol{K}_s \mid \boldsymbol{B}_r,\boldsymbol{B}_s,\boldsymbol{S}_i)=0 \tag{4.4.6}$$

定义 4.4.1 的性质（1）到性质（4）完整地继承了定义 4.3.5 的性质；而定义 4.4.1 中的性质（5）则描述了 $\boldsymbol{D}_S(\boldsymbol{U},t,m)$是一个自愈的组密钥分发模型所应满足的属性。同样，考虑到组密钥分发协议的安全性需求，$\boldsymbol{D}_S(\boldsymbol{U},t,m)$还应满足组密钥的前向隐私性和后向隐私性（见定义 4.3.6）。

4.4.2 组密钥的自愈机制和后向隐私机制

S-GKDS 组密钥的自愈机制和后向隐私机制是基于单向哈希链的，它依赖于基本的单向哈希函数。

单向哈希函数以一个变长的二进制串 M 作为输入，并输出一个固定长度的散列二进制串 $H(M)$。一个单向函数 H 满足以下两个性质：①给定 x，能够计算出 y，使得 $y=H(x)$；②给定 y，不能计算出 x，使得 $y=H(x)$。如哈希散列函数就是典型的单向函数。

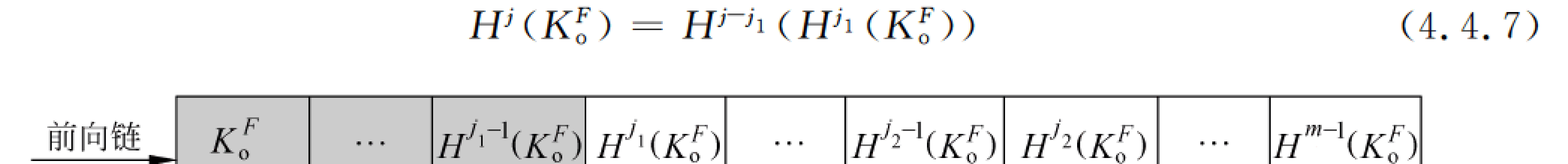
一个单向哈希链则是一个哈希值序列$\{x_n,\cdots,x_j,\cdots,x_0\}$，且满足$\{x_j \mid \forall j:0<j\leqslant m, x_{j-1}=H(x_j)\}$，其中 x_n 是单向哈希链的一个秘密种子，哈希序列值之间满足如下线性关系：$x_1=H(x_2)=\cdots=H^{m-2}(x_{m-1})=H^{m-1}(x_m)(1<j\leqslant m)$。由于哈希函数 H 的单向性，给定 x_i，则不可能计算出 $x_j(j<i)$，然而，通过 $x_j=H^{j-i}(x_i)$，则容易计算出 $x_j(j>i)$。

4.4.2.1 组密钥的后向隐私性

为了保证组密钥的后向隐私性，我们在密钥分发协议 S-GKDS 中引入一个前向哈希链 K^F。该单向哈希链通过对一个秘密种子重复应用一个单向哈希函数 H 来产生，即：①产生一个随机密钥种子值 K_o^F；②对种子 K_o^F 重复使用单向函数 H 产生一个长度为 m 的哈希链：$\{K_o^F,H(K_o^F),\cdots,H^{m-1}(K_o^F)\}$。这种产生机制类似于 S/KEY[66]。

如图 4.4.1 所示，前向哈希链保证了组密钥的后向隐私性。设 m 是组会话的最大次数。当一个用户在会话 j_1 加入到一个活动组时，GC 将与会话 j_1 对应的哈希链上的值 $H^{j_1}(K_o^F)$预先分发给这个新的用户。显然，对在会话 j_1 加入活动组的新用户而言，由于哈希函数的单向性，它难以计算在会话 j_1 以前的哈希值序列$\{H^j(K_o^F)\mid 1\leqslant j<j_1\}$；而对会话 j_1 以后(含会话 j_1)的哈希链而言，组用户则能使用预分配的哈希值 $H^{j_1}(K_o^F)$来计算在 $j_1\leqslant j\leqslant m$ 范围的哈希序列$\{H^j(K_o^F)\mid j_1\leqslant j\leqslant m\}$：

$$H^j(K_o^F)=H^{j-j_1}(H^{j_1}(K_o^F)) \tag{4.4.7}$$



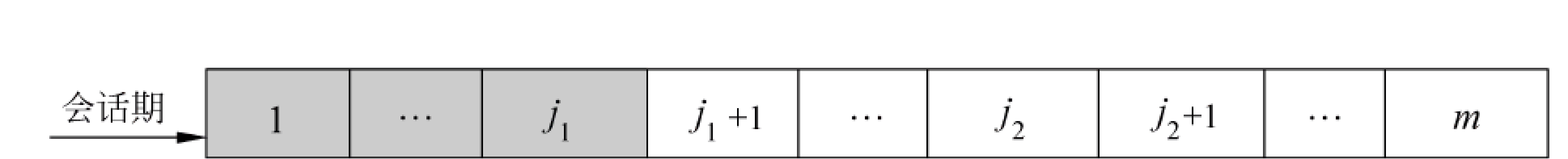


图 4.4.1 组密钥的后向隐私性：基于单向哈希链的方法

在 S-GKDS 协议中，最大会话次数为 m 的通信组在会话 j 的组通信密钥被定义为 j，$H^j(K_o^F)$和 RK_j 的函数值：

$$TEK_j=f(H^j(K_o^F),RK_j) \tag{4.4.8}$$

其中，$1\leqslant j\leqslant m$；K_o^F 是前向哈希链的种子值；函数 $f(\cdot)$也是一单向函数，它能把任意长度的消息经过处理后输出为一个固定长度的值；RK_j 来源于 GC 的第 j 次组密钥更新消息，其更新处理机制将在随后的章节中详细介绍。

由组通信密钥的构造方式可知，组密钥的后向隐私性能得以有效保证。由于前向哈希

链 K^F 的单向性，具有 $H^{j_1}(K_0^F)$ 的组用户只能访问在 $j_1\leqslant j\leqslant m$ 范围内的组通信密钥，因为它能利用预分配的哈希值 $H^{j_1}(K_0^F)$ 来计算 $\{H^j(K_0^F)|j_1\leqslant j\leqslant m\}$，即 $H^j(K_0^F)=H^{j-j_1}(H^{j_1}(K_0^F))$；而对于会话 j_1 前的组通信密钥而言，组用户难以计算出 $H^j(K_0^F)$，从而不可能计算出在 $1\leqslant j<j_1$ 范围内的组通信密钥 TEK_j。

值得一提的是，在 S-GKDS 协议中，组密钥的后向隐私性则完全依赖于组密钥更新消息，具体的机制在后续部分将详细介绍。

4.4.2.2 组密钥更新消息的自愈机制

实际上，单向哈希链还能为组密钥更新消息的分发提供一种有效的自愈机制。正如上文所提到的，对任何一个在会话 $j(1\leqslant j\leqslant m)$ 的活动用户而言，其都能够有效地计算出第 j 次的组通信密钥 $\text{TEK}_j=f(H^j(K_0^F),\text{RK}_j)$。其中，$H^j(K_0^F)$ 并不需要在密钥更新消息中传输，因为每一个用户都能利用预先设置的哈希值 $H^{j_1}(K_0^F)$ 独立计算出 $H^j(K_0^F)=H^{j-j_1}(H^{j_1}(K_0^F))$；而 RK_j 则需要被封装在密钥更新消息中，由 GC 周期性地以广播方式分发到所有活动用户。

因此，组密钥的自愈机制要求协议提供一种强健的机制，使得密钥更新消息 RK 能在不可靠的广播信道上可靠地进行传输。为此，在初始阶段，GC 选择一个随机数 RK_m 作为哈希链的秘密种子值，并使用 RK_m 重复执行哈希函数 H 来预先计算出单向哈希链 $\{\text{RK}_i|i=1,2,\cdots,m\}$，其中 $\text{RK}_i=H(\text{RK}_{i+1})$，$1\leqslant i\leqslant m-1$。最后，所有的更新消息 RK 构成以下线性关系：

$$\text{RK}_1 = H(\text{RK}_2) = \cdots = H^{m-2}(\text{RK}_{m-1}) = H^{m-1}(\text{RK}_m) \tag{4.4.9}$$

在以后的组密钥更新阶段，GC 将 $\text{RK}_i(i=1,2,\cdots,m)$ 按倒序逐步分发给所有的活动用户，即在会话 0 释放 RK_0，在会话 1 释放 RK_1，…，在会话 m 释放 RK_m。一旦获得会话 j 释放的更新消息 RK_j，用户则能够方便地利用单向函数 H 来恢复先前丢失的密钥更新消息 $\{\text{RK}_i|\text{RK}_i=H^{j-i}(\text{RK}_j),1\leqslant i\leqslant j\}$。值得指出的是，这种自愈机制在一定程度上还保证了组密钥的后向隐私性。这是因为，用户难以计算在会话 j 后其他的更新消息 $\{\text{RK}_i|j+1\leqslant i\leqslant m\}$。

因此，尽管密钥更新消息很可能在传输过程中丢失，但用户仍然可以通过哈希函数和最近接收到的 RK 来恢复在以前的更新消息中丢失的那些 RK。后续的性能分析将表明这种自愈机制能够有效地容忍信道的高丢包率和错误率。

4.4.3 自愈的组密钥分发协议

自愈的组密钥分发协议 S-GKDS 也属于集中式的安全组密钥管理方案。在该协议中，由组控制器 GC 担当中央控制节点来负责组密钥的创建、分发和组成员关系发生变化时的组密钥更新。前述 B-GKDS 协议所关注的主题是协议的动态稳定性，而 S-GKDS 协议重点解决的主题是协议的鲁棒性。

S-GKDS 协议几乎完全是 B-GKDS 协议的扩展，它完整地继承了 S-GKDS 协议的基本特性，如安全性和组用户动态参与机制的灵活性；并通过引入一种基于单向哈希链的组密钥自愈机制来保证协议的鲁棒性。这种组密钥自愈算法具有较小的计算和通信开销。它使得用户节点仅需要进行快捷的哈希计算就能保证组密钥更新消息的可靠传输和组密钥更

新效率，进而使得 S-GKDS 协议对丢包率和错码率较高的组通信环境具有更好的自适应性。

下面我们将依次讨论 S-GKDS 协议的基本体系结构、活动组用户的初始化、组密钥的更新和恢复机制、容忍时钟扭曲的机制以及组用户的动态参与特性。

4.4.3.1 S-GKDS 的体系结构

与 B-GKDS 协议一样，我们假定组通信系统的组成员和 GC 之间有一个统一的全局时钟(可容许各节点和 GC 之间存在时钟漂移)。令 m 是组通信系统的活动周期。相应地，组通信系统的活动周期将被划分成 m 次会话。

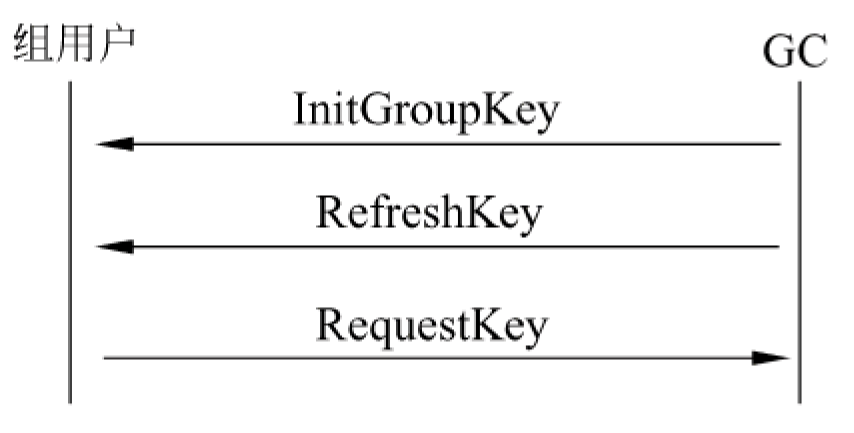


图 4.4.2　S-GKDS 协议中 GC 和组用户之间的消息流

如图 4.4.2 所示，在 S-GKDS 协议中包括有 3 种类型的消息：InitGroupKey，RequestKey 和 RefreshKey。InitGroupKey 和 RequestKey 是 GC 和单个组用户之间的单播，而 RefreshKey 被广播到所有的组用户。InitGroupKey 消息用来初始化密钥更新参数，在初始化阶段被发布到用户。RefreshKey 消息用来周期性地广播密钥更新消息到所有用户。如果某个组用户没有收密钥更新消息的时间超过一个预先设置的更新间隔 b，则产生 RequestKey 消息以向 GC 明确请求密钥更新消息。

图 4.4.3 描述了 S-GKDS 协议所包含的主要 4 个功能模型：消息包的验证和检查模块、组密钥更新消息的自愈模块、组密钥 TEK 切换模块以及流加密/解密和完整性检查模块。消息包的验证模块(message verification module)着力于解决类似 DoS 类型的攻击，使得系统在处理组密钥更新消息 RefreshKey 时能够在一定程度上抵御 DoS 类型的攻击。这里需要引入一种有效的包过滤技术：当接收到 RefreshKey 消息时，每个用户用以前保留的 RK 来隐性验证接收到的 RK，而不依赖于 GC 重传丢失的 RK。

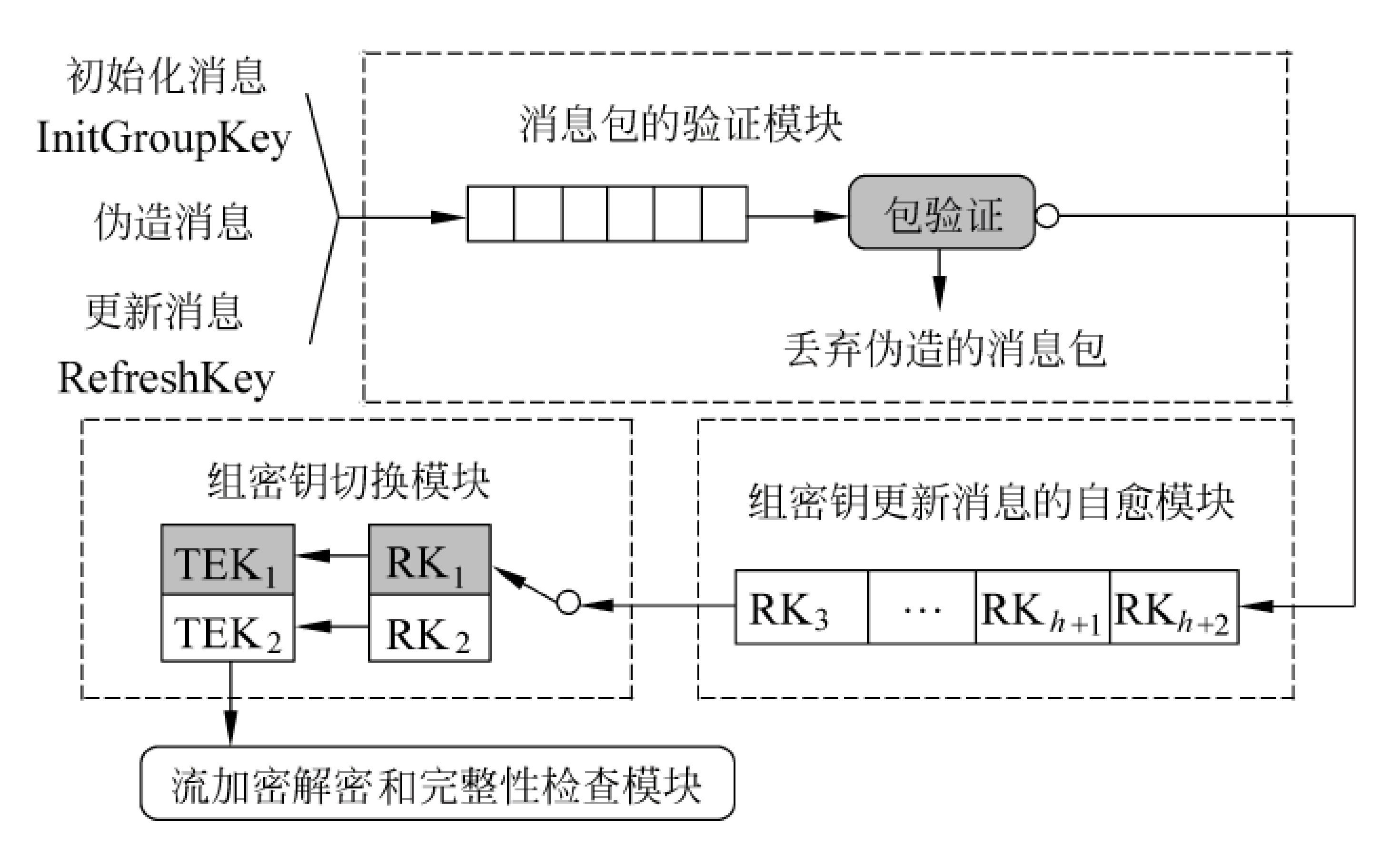


图 4.4.3　S-GKDS 协议的体系结构

组密钥更新消息的自愈模块(self-healing module)为容忍广播通道上的丢包提供了密钥更新消息的自恢复机制。尽管密钥更新消息 RefreshKey 很可能在传输过程中丢失，但用户仍然可以利用最近接收到的 RK 来通过计算恢复在以前丢失的 RK，而不需要请求 GC 重

传丢失的 RK。这种自愈机制依赖于哈希函数的单向性，相似的机制也出现在 TESLA[67,68] 和 LISP[69] 中。本文提出的算法具有：①更好的效率，因为每个组用户只需要存储固定数量的更新消息，而 TESLA 则要求存储所有接收到的消息，直到用户接收了一个认证消息；②高效的用户加入和退出机制，这一点在相关协议中未曾得以有效地考虑。

组密钥切换模块(traffic encryption key switch module)可以无缝地切换组通信密钥，而不干扰正在传输的数据。图中的两个密钥槽可以同步进行操作。当一个密钥槽中的 RK 被用来加密或解密数据时，接收到的新 RK 将被写入到另一个密钥槽中。在密钥更新点上，用户将活动密钥槽切换到另一个，并接收新的 RK。

4.4.3.2 S-GKDS 组用户的初始化

与 B-GKDS 协议相类似，在 S-GKDS 组用户的初始化阶段，组管理中心 GC 同样需要从 $F_q[x]$ 中随机地选取 m 个度为 t 的屏蔽多项式：

$$\{h_j(x) = h_{0,j} + h_{1,j}x + \cdots + h_{t,j}x^t\}_{j=1,2,\cdots,m} \in F_q[x] \tag{4.4.10}$$

另外，为了保证组密钥的后向隐私性，S-GKDS 的 GC 需预先计算一个前向哈希链 $K^F=\{K_o^F, H(K_o^F), \cdots, H^i(K_o^F), \cdots, H^m(K_o^F)\}$；为了提供密钥更新消息的自愈机制，GC 还需预先计算一个密钥更新消息 RK 的单向哈希序列$\{\mathrm{RK}_i \mid i=1,2,\cdots,m\}$，以满足 $\mathrm{RK}_i = H(\mathrm{RK}_{i+1})$，$0 \leqslant i \leqslant m-1$。

随后，GC 利用多项式$\{h_j(x)\}_{j=1,2,\cdots,m}$为每个在初始化阶段加入活动组的用户 U_i 产生 m 个秘密私钥 $S_i=\{h_1(i), h_2(i), \cdots, h_m(i)\}$，并通过 InitGroupKey 消息将这 m 个秘密私钥$\{h_1(i), h_2(i), \cdots, h_m(i)\}$以及与会话 1 对应的哈希链 K^F 上的值 $H(K_o^F)$通过安全可靠的信道预先分发给用户 U_i。

$$\mathrm{GC} \rightarrow U_i: \{E_{\mathrm{MK}_i}(b \mid \mathrm{RK}_{b+2} \mid T_{\mathrm{refresh}} \mid H(K_o^F) \mid h_1(i) \mid \cdots \mid h_m(i)) \mid$$
$$\mathrm{MAC}(b \mid \mathrm{RK}_{b+2} \mid T_{\mathrm{refresh}} \mid H(K_o^F) \mid h_1(i) \mid \cdots \mid h_m(i))\},$$

其中，b 是密钥更新消息 RK 的缓冲区长度；T_{refresh}是密钥更新周期，即组会话间隔时间；MAC(·)是产生消息验证码的散列函数(如 SHA-1，MD5)；MK_i 是用户 U_i 与 GC 共享的主密钥，用于 InitGroupKey 消息的加密和验证。

U_i 一旦接收到 InitGroupKey 消息，首先验证该消息的真伪；若为真，则 U_i 利用 MK_i 解密该消息并获得相应的秘密私钥 $S_i=\{h_1(i), h_2(i), \cdots, h_m(i)\}$、哈希链 K^F 上的哈希值 $H(K_o^F)$以及密钥更新消息 RK_{b+2}。随后，用户利用单向函数 H 计算其他的哈希序列值$\{\mathrm{RK}_j\}_{j=1,2,\cdots,b+1}$；复制 RK 消息序列到它对应的密钥更新消息缓冲区 kb 和密钥更新消息槽 ks；计算组密钥$\{\mathrm{TEK}_j = f(H^j(K_o^F), \mathrm{RK}_j)\}_{j=1,2}$并复制到对应的组密钥槽 tb 中；最后指定 TEK_1 为当前活动的组密钥。组用户完成这些初始化处理后，可通过接收随后的密钥更新消息 RefreshKey 来同步地更新组密钥。详细的组用户初始化和组密钥的更新机制可参见图 4.4.4 和算法 4.4.1。

此外，如果用户的计时器到期，则算法 4.4.2 被触发执行。算法 4.4.2 处理组密钥更新消息 RK 的右移操作以便 RK 在每个间隔 T_{refresh}都可以自动更新。这里，用户设置一个变量 e 用于统计没有被用户正确接收的密钥更新消息 RefreshKey 数目；当 $e=b$ 时，则表明该用户没有收到密钥更新消息的次数已超过一个预先设置的门限值 b，这时，用户需产生一个 RequestKey 消息以向 GC 明确请求组密钥更新消息。

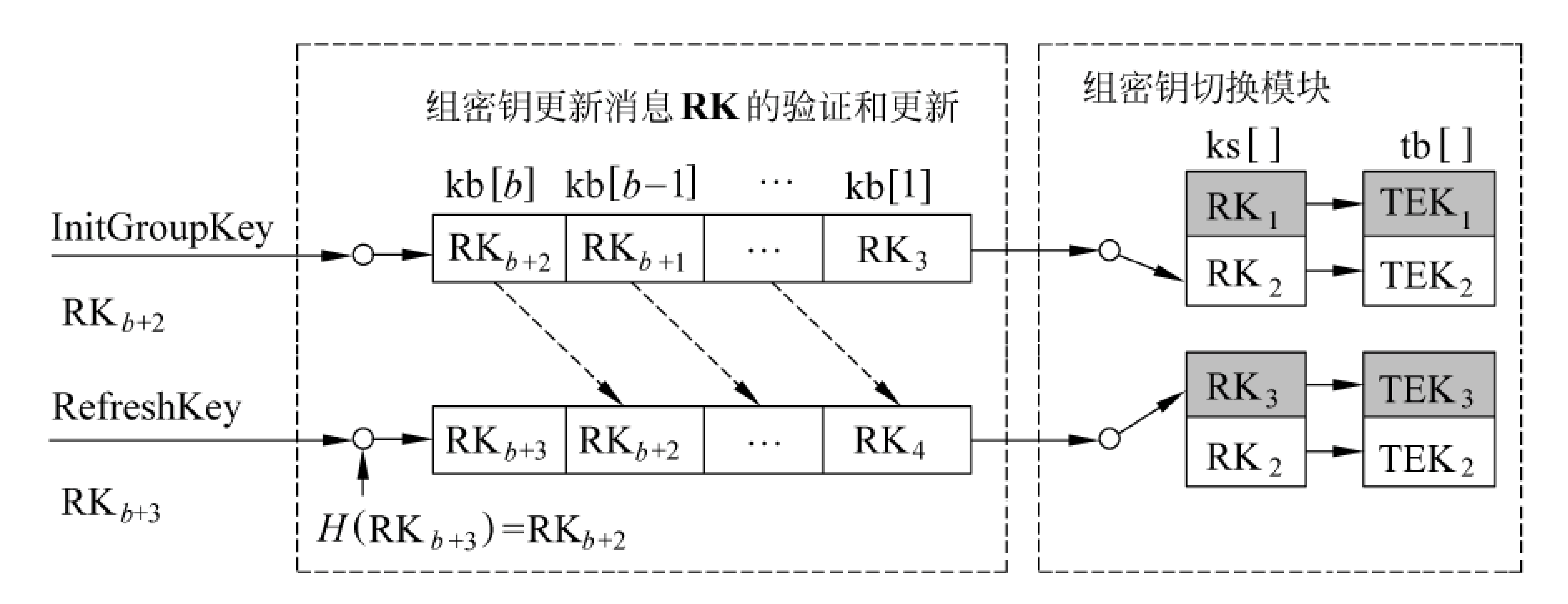


图 4.4.4　组用户的初始化和组密钥的更新

算法 4.4.1　组用户的初始化

Function Init_TEK(){
　if (接收到 **InitGroupKey** 消息){
　　解密 **InitGroupKey** 消息得到{RK_{b+2}, b, $T_{refresh}$, $H(K_o^F)$, $h_1(i)$, …, $h_m(i)$};
　　分配一个长度为 b 的密钥更新消息缓冲区{kb[1], …, kb[b]};
　　分配一个密钥更新消息槽{ks[1], ks[2]}和与之对应的组密钥槽{tk[1], tk[2]};
　　for ($i=1; i<=b; i++$) **do** kb[i]$=H^{b-i}(RK_{b+2})$;
　　ks[2]$=H^b(RK_{b+2})$; ks[1]$=H^{b+1}(RK_{b+2})$;
　　tk[2]$=f($ks[2]$, H^2(K_o^F))=f(H^b(RK_{b+2}), H^2(K_o^F))$;
　　tk[1]$=f($ks[1]$, H^1(K_o^F))=f(H^{b+1}(RK_{b+2}), H^1(K_o^F))$;
　　$RK_w=$ ks[2]$=H^b(RK_{b+2})$;
　　设置 tk[1]中的组密钥为当前活动的组密钥;
　　设置 RefreshKeyTimer 为 $T_{refresh}/2$;
　　$e=0$; /* 表示没有接收到的更新消息 **RefreshKey** 数目 */
　}
}

算法 4.4.2　更新密钥计数器

Function Refresh_Key_Timer(){
　if (RefreshKeyTimer 被触发){/* 组系统正处于会话 j 阶段 */
　　kb[1]→ks[(j mod 2)+1]; /* 右移 kb[1]到当前非活跃的密钥槽 */
　　tk[(j mod 2)+1]$=f(H^j(K_o^F)$, ks[(j mod 2)+1]); /* 计算非活跃密钥槽中的 TEK */
　　for ($i=1; i<=b-1; i++$) **do** kb[i]→kb[$i-1$]; /* 右移组密钥更新消息 RK */
　　设定密钥槽 tk[j mod 2]中组密钥的活跃组密钥;
　　重新设定 RefreshKeyTimer 为 $T_{refresh}$;
　　$e++$;
　　if ($e==b$)发送 **RequestKey** 消息到 GC;
　}
}

4.4.3.3 S-GKDS 组密钥的更新机制

初始化后，GC 将周期性地将密钥更新消息$\{RK_i \mid RK_{i-1}=H(RK_i), i=1,2,\cdots,m\}$按逆序依次分发给所有的活动用户。考虑 GC 的第 j 次组密钥更新，设 R_j 表示在会话 j 被撤销的组用户集，则 $R=R_j \cup R_{j-1} \cup \cdots \cup R_1$（$|R| \leqslant t, R \cap G_j=\varnothing$）表示在会话 j 前被撤销的所有组用户集。为保证在广播信道上安全地传输密钥更新消息，以满足任意的活动组用户 $U_i \in G_i$ 能够解密这个消息，而任何 R 中的非活动用户（$|R| \leqslant t$）即使他们以任意的模式密谋也不能解密该消息。为此，对于通信组的第 j 次密钥更新，GC 将广播如下的组密钥更新消息 RefreshKey（或称为 B_j）给所有活动用户：

$$GC \rightarrow *: \{w_j(x) \mid \{R\} \mid MAC(\{R\} \mid w_j(x))\},$$

其中，$w_j(x)=g_j(x) \cdot RK_j+h_j(x)$是一个度为 t 的多项式，$h_j(x)$是屏蔽多项式；RK_j 是当前的组密钥更新消息；多项式 $g_j(x)$则按如下方式构造：

$$g_j(x)=\prod_{r_i \in R}(x-r_i) \tag{4.4.11}$$

4.4.3.4 S-GKDS 组密钥的恢复机制

S-GKDS 组密钥的恢复机制与 B-GKDS 协议相类似，算法 4.4.3 详细描述了组用户对消息组更新消息 RefreshKey 的处理过程。对某一特定的活动用户 $U_i \in G_i$ 而言，当收到 GC 广播的组密钥更新消息 B_j 时，它首先计算多项式 $w_j(x)$和 $g_j(x)$在点 i 处的值 $w_j(i)$和 $g_j(i)$。由于 $U_i \notin R$，故有 $g_j(i) \neq 0$，因此，用户 U_i 可以利用它在初始化阶段保留的秘密私钥 $h_j(i)$以及 $w_j(i)$和 $g_j(i)$进一步恢复出当前的组密钥更新消息 RK_j。

算法 4.4.3 组密钥更新消息 RK 和组密钥 TEK 的计算

```
Function TEK_Refresh_Recover(){
  while (活动用户 U_i∈G_i 接收到 RefreshKey 消息 B_j){
    利用消息 B_j 中的 R 构造多项式 g_j(x)= ∏_{r_i∈R} (x－r_i) ;
    计算多项式 w_j(x)和 g_j(x)在点 i 处的值 w_j(i)和 g_j(i) ;
    RK_j=(w_j(i)－h_j(i))/g_j(i) ;
    if (H^{e+1}(RK_j)≠kb[b－e]){/ * 验证消息包是否正确 * /
      丢弃当前的 RefreshKey 消息 ;
      continue;
    }
    kb[1]→ks[(j mod 2)+1];/ * 右移 kb[1]到当前非活跃的密钥槽 * /
    tk[(j mod 2)+1]=f(H^j(K_0^F),kb[1]);/ * 计算非活跃密钥槽中的组密钥 TEK * /
    for (i=1;i<=b－1;i++)do kb[i]→kb[i－1] ;/ * 右移组密钥更新消息 RK * /
    设定密钥槽 tk[j mod 2]中组密钥为当前活跃的组密钥;
    if (e≠0){ / * 有丢失的 RKs * /
      for (k=0;k<e;k++) do / * 恢复丢失的 RKs * /
        H^k(RK_j)→kb[t－k] ;
      e=0 ; / * 重置值 e * /
    }
  }
}
```

$$\mathrm{RK}_j = (w_j(i) - h_j(i))/g_j(i) \tag{4.4.12}$$

随后，活动用户 $U_i \in G_i$ 可按下式计算当前的组通信密钥：

$$\mathrm{TEK}_j = f(H^j(K_o^F), \mathrm{RK}_j) \tag{4.4.13}$$

另一方面，对于 $R=R_j \cup R_{j-1} \cup \cdots \cup R_1$（$|R| \leqslant t$）中被 GC 撤销的用户 U_r 而言，由于 $\{g_j(r)=0 \mid \forall U_r \in R\}$，$w_j(r)=g_j(r) \cdot \mathrm{RK}_j + h_j(r) = h_j(r)$，因此，即使他们密谋（$|R| \leqslant t$）也难以计算出当前的组更新消息 RK_j；进而，也难以计算与之对应的组通信密钥 $\mathrm{TEK}_j = f(H^j(K_o^F), \mathrm{RK}_j)$。

组密钥 TEK 可以和最新的密钥更新消息进行同步更新（算法 4.4.3 的第 11 行）。由于 RK 序列的单向性，RefreshKey 消息并不需要消息认证码，因为接收者通过检测 $H^{e+1}(\mathrm{RK}_j) \neq \mathrm{kb}[b-e]$ 来验证 RK 是否属于相同的组密钥更新消息序列（算法 4.4.3 的第 6 行）。这种隐性认证的方法很显著地减少了广播消息的大小。

由于仅需要处理低开销的哈希操作，因此组密钥更新操作的计算开销并不大。同时，组密钥更新消息的隐性认证机制使得 GC 和用户之间无需消息重传，这也极大地减少了协议通信开销。

4.4.3.5 S-GKDS 组密钥更新消息的自愈机制

尽管由于广播信道的不可靠性，GC 在分发组密钥更新消息 RefreshKey 给组用户时，可能会出现丢包现象；但用户仍然可以通过哈希函数 H 并利用最近接收到的 RK 来恢复在先前更新消息中丢失的那些 RK。

事实上，密钥更新消息 RefreshKey 提供了一种简洁的自愈机制来恢复丢失的 RK。假设由于消息的丢失仅有 $r(\leqslant b)$ 个 RK 被保留在密钥缓冲区内，因此缓冲区有 $e=b-r$ 个空槽。令 $\{\mathrm{RK}'_r, \mathrm{RK}'_{r-1}, \cdots, \mathrm{RK}'_1\}$ 为缓冲区 $\{\mathrm{kb}[r], \cdots, \mathrm{kb}[1]\}$ 的 r 个 RK，属于相同的 RK 序列，且满足 $H(\mathrm{RK}'_r)=\mathrm{RK}'_{r-1}, \cdots, H(\mathrm{RK}'_2)=\mathrm{RK}'_1$。

算法 4.4.3 给出的自愈算法（14～18 行）能够有效地恢复这些空槽中的密钥更新消息 RK：当收到最新的 RefreshKey 消息 RK_k 时，每个用户首先检查是否有 $H^{e+1}(\mathrm{RK}_{i+1}) == \mathrm{kb}[b-e]$；如果成立，则用当前的 RK 恢复相同密钥更新序列中丢失的 RK。图 4.4.5 也说明这种自愈机制。假设一个用户收到 RefreshKey 消息，如果 $H(\mathrm{RK}_{b+3})=\mathrm{RK}_{b+2}$ 和 $e=0$，则消息包的验证通过，表明用户收到一个合法的更新消息包；但在接下来的两个密钥更新周期，因为密钥更新消息由于验证错误（$H(\mathrm{RK}^*_{b+4}) \neq \mathrm{RK}_{b+3}$，$H^2(\mathrm{RK}^*_{b+5}) \neq \mathrm{RK}_{b+3}$）而被丢弃，$e=2$；随后，当接收到下一个 RefreshKey 消息 RK_{b+6} 时，因为 $H^3(\mathrm{RK}_{b+6})=\mathrm{RK}_{b+3}$，则活动用户可以成功地恢复出以前丢失的两个 RK，即 $\mathrm{RK}_{b+4}=H^2(\mathrm{RK}_{b+6})$ 和 $\mathrm{RK}_{b+3}=H^3(\mathrm{RK}_{t+6})$。

在恢复出丢失的 RK 后，组活动用户可以成功地计算出先前无法获得的组通信密钥 $\mathrm{TEK}_j = f(H^j(K_o^F), \mathrm{RK}_j)$。

4.4.3.6 组用户的动态参与机制

组用户的动态参与机制要求协议在用户加入或离开活动组的情况下，能够有效地保证组会话密钥的前向隐私性和后向隐私性。

用户离开：S-GKDS 的组用户撤销机制与 B-GKDS 基本一致，是通过组密钥的更新消

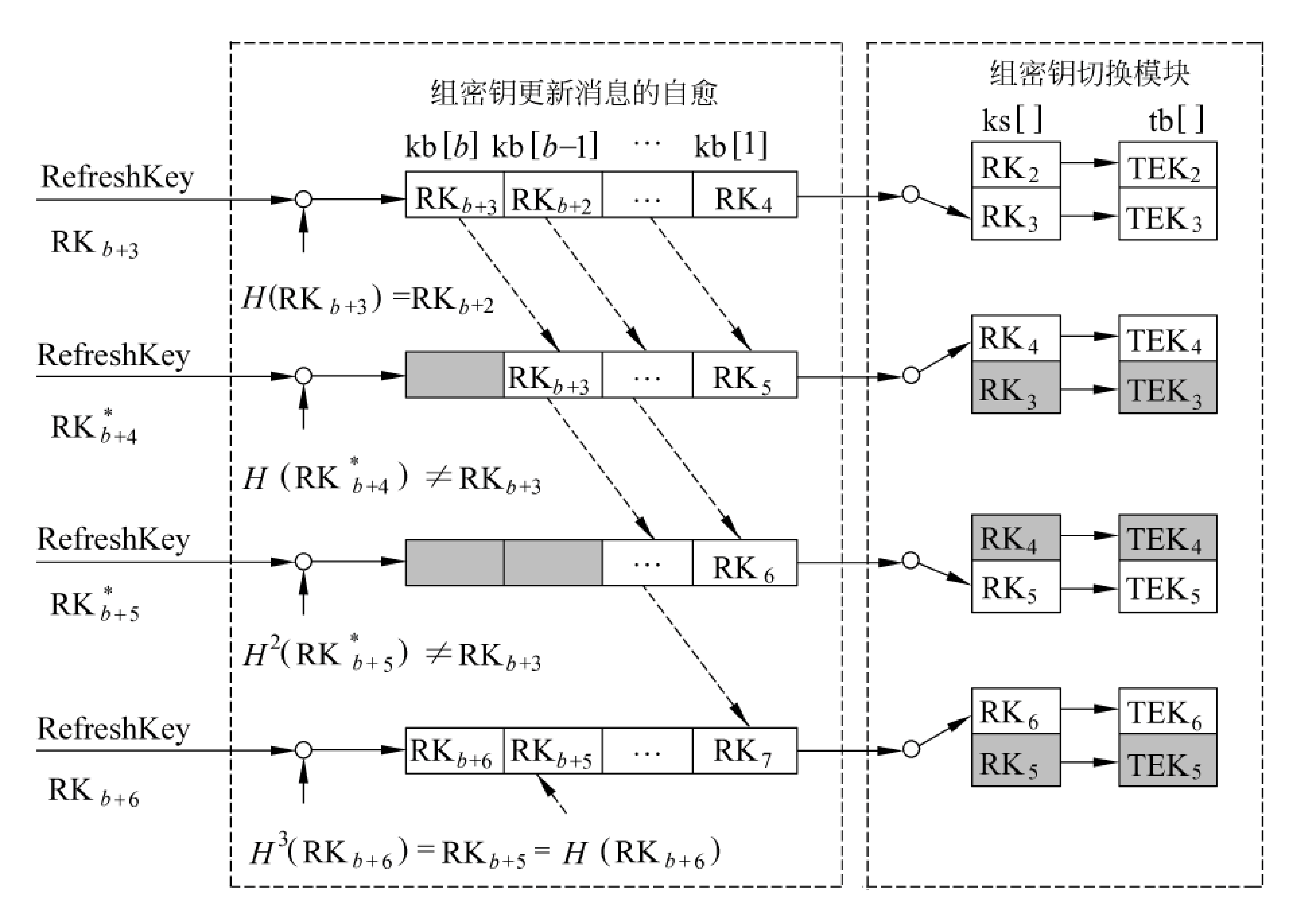


图 4.4.5 组密钥更新消息 RK 的自愈机制

息来实现的。假设在第 j 次会话中，GC 需要撤销用户 U_r，则只需要 U_r 包括在广播消息 B_j 的$\{R\}$集合中，即 $U_r \in R$。由于$\{g_j(r)=0 \mid \forall U_r \in R\}$，用户 U_r 无法利用广播的密钥更新消息 B_j 及其自己先前保存的秘密 $h_j(r)$去计算当前的组密钥更新消息 RK_j，自然也无法计算组密钥 TEK_j。

用户加入：当用户 U_v 希望在会话 j 加入活动组时，其相应的处理与组用户在 S-GKDS 协议初启时的初始化过程相类似，即：

(1) 用户 U_v 首先需要从 GC 获得加入活动组的许可；如果成功，U_v 建立一个与 GC 共享的主密钥 MK_i。随后 GC 为 U_v 产生 $m-j+1$ 个秘密私钥$\{h_j(v), h_{j+1}(v), \cdots, h_m(v)\}$，并通过 InitGroupKey 消息将秘密私钥$\{h_j(v), h_{j+1}(v), \cdots, h_m(v)\}$和与会话 j 相对应的哈希链 K^F 上的值 $H^j(K_o^F)$通过安全、可靠的信道分发给用户 U_v：

$$GC \rightarrow U_v: \{E_{\mathrm{MK}_v}(b \mid \mathrm{RK}_{b+2} \mid T_{\mathrm{refresh}} \mid H^j(K_o^F) \mid h_j(v) \mid \cdots \mid h_m(v)) \mid$$
$$\mathrm{MAC}(b \mid \mathrm{RK}_{b+2} \mid T_{\mathrm{refresh}} \mid H^j(K_o^F) \mid h_j(v) \mid \cdots \mid h_m(v))\},$$

其中，共享的主密钥 MK_v 用于 InitGroupKey 消息的加密和验证；$H^j(K_o^F)$是 U_v 在前向哈希链中与会话 j 对应的哈希值；b 是密钥更新消息 RK 的缓冲区长度；T_{refresh}是密钥更新周期。

(2) 用户 U_v 一旦接收到 InitGroupKey 消息，则按照算法 4.4.1 处理该消息，然后加入到活动组通信，接收随后的密钥更新消息 RefreshKey 并同步地更新组密钥 TEK，如算法 4.4.3 所示。

4.4.3.7 组用户的重新初始化

组用户可能因 RK 的认证无效或者组密钥更新消息 RefreshKey 的丢失，而导致其组密钥消息缓冲区里所有 b 个槽全为空，即 $k=b$；这时，GC 需要重新对该用户进行初始化，具体

步骤为：

（1）组用户首先向 GC 发送 RequestKey 消息明确要求获得当前的组密钥更新消息；

（2）GC 将当前的组密钥更新参数封装在 InitGroupKey 消息内，并以单播的方式发送给请求该服务的用户；

（3）一旦用户接收到新的初始化消息 InitGroupKey，它则按算法 4.4.1 来处理该消息。随后，该用户便可以通过接收 RefreshKey 消息来周期性地更新组密钥 TEK。

4.4.3.8 时钟扭曲（clock skews）

S-GKDS 协议能够在一定程度上容忍组通信中各组件（用户节点和 GC）之间的时钟不同步，即协议并不要求用户节点和 GC 之间的时钟完全精确同步。这使得 S-GKDS 协议对某些时延较大的通信环境具有良好的自适应性。

令 $m_i(T)$表示在第 i 次组会话时组成员 U_i 对应的全局时钟，T 是组用户节点的时钟。则组成员 A 和 B 之间的时钟扭曲可表示为$\lambda = |m_A(T) - m_B(T)|$。

为满足无缝地更新组密钥，λ 应该满足 $\lambda < T_{\text{refresh}}/2$，因为根据算法 4.4.1 和算法 4.4.2，用户会在密钥更新周期的中间时刻将活动的组密钥自动地切换到密钥槽中的另一个组密钥。

假定系统的组会话 1 始于时刻 t_{init}。在协议的第 i 次组密钥更新，若在节点 A 和 B 之间存在时钟扭曲，即用户节点 A 使用 TEK_{i-1}作为组通信密钥，而节点 B 使用 TEK_i。显然，在此情形下，节点 A 和 B 之间仍然能够在该更新周期内正常地进行通信，因为 A 和 B 同时保留有组密钥对$\{\text{TEK}_{i-1},\text{TEK}_i\}$。

考虑到用户的计时器的更新周期是 $T_{\text{refresh}}/2$，即每隔 $T_{\text{refresh}}/2$，算法 4.4.2 将被触发执行，执行结果将导致组节点将活动的组密钥自动地切换到密钥槽中的另一个组密钥。因此，在最坏的情形下，即当组节点之间的时钟扭曲达 $T_{\text{refresh}}/2$ 时，S-GKDS 协议仍然能够保持正常的通信。即对任何两个组用户节点 A 和 B 而言，可容忍的时钟扭曲边界值是：

$$\max\{|m_A(T) - m_B(T)|, \forall A, B \in \boldsymbol{U}\} < T_{\text{refresh}}/2 \tag{4.4.14}$$

另外，GC 和组用户之间也能容忍一定的时钟扭曲。考虑到第 i 次组密钥更新，GC 将在时刻 $t_{\text{init}} + i \cdot T_{\text{refresh}}$广播更新消息 RefreshKey 给各节点。同样，GC 和组节点之间所能容忍的时间范畴为$[t_{\text{init}} + (i-1/2) \cdot T_{\text{refresh}}, t_{\text{init}} + (i+1/2) \cdot T_{\text{refresh}}]$，即所能容忍的最大时间扭曲值是：

$$\max\{|m_{GC}(T) - m_A(T)|, \forall A \in \boldsymbol{U}\} < T_{\text{refresh}}/2 \tag{4.4.15}$$

4.4.4 安全性分析

这里，我们对 S-GKDS 协议进行与 B-GKDS 相类似的安全性分析。同样，在如下定理的推导过程中，我们用随机变量 $\mathbf{K}_j (j=1,2,\cdots,m)$表示对应的组会话密钥 $\text{TEK}_i (i=1,2,\cdots,m)$；用随机变量 $\mathbf{RK}_j (j=1,2,\cdots,m)$表示对应的组密钥更新消息 $\text{RK}_i (i=1,2,\cdots,m)$。

定理 4.4.1 组密钥管理协议 S-GKDS 能同时安全地撤销最多 t 个用户；并且，就信息论范畴而言，S-GKDS 协议是一个无条件安全的自愈组密钥分发协议。

证明：依据定义 4.4.1，假定 $\boldsymbol{U}=\{U_1,U_2,\cdots,U_n\}$是所有可能的组用户集；$R_j \subseteq \boldsymbol{U}$ 是第 j 次会话中由 GC 所撤销的组用户；$R = R_j \cup R_{j-1} \cup \cdots \cup R_1 (|R| \leqslant t)$是在会话 j 被撤销的所有

组用户集；$J_j \subseteq \boldsymbol{U}$ 是第 j 次会话中新加入的组用户；$G_j=(G_{j-1}\cup J_j)\backslash R_j$ 是第 j 次会话中合法的组成员；m 表示组通信系统的最大会话次数；t 是 GC 所能撤销的最大用户数。

对比定义 4.3.1 和定义 4.3.5 可知，自愈的组密钥分发模型 $\boldsymbol{D}_S(U,t,m)$ 的性质 4.3.1～性质 4.3.4 完整地继承了 $\boldsymbol{D}_B(\boldsymbol{U},t,m)$ 的特性；并在此基础上，定义 4.3.1 引入了组密钥的自愈特性。S-GKDS 协议则完全是 B-GKDS 协议的扩展，它具有 B-GKDS 协议的安全特性。因此，S-GKDS 协议满足 $\boldsymbol{D}_S(\boldsymbol{U},t,m)$ 模型所定义的性质 4.3.1～性质 4.3.4。下面只需证明 S-GKDS协议能够满足 $\boldsymbol{D}_S(\boldsymbol{U},t,m)$ 所定义的性质 4.3.5 即可。

对某一特定的活动用户 $U_i\in G_r$ 而言，若其从会话 r 开始一直到随后的会话 $s-1$ 期间一直都未收到组密钥更新消息 $\{B_i | r<i\leqslant s-1\}$；但一旦收到会话 s 期间的组密钥更新消息 B_s，它便能利用组密钥的自愈机制恢复在 (r,s) 期间丢失的组密钥 $\{K_i | r<i<s\}$。具体而言，对 $U_i\in G_i$，当它收到 GC 广播的密钥更新消息 B_s 时，它首先计算多项式 $w_s(x)$ 和 $g_s(x)$ 在点 i 处的值 $w_s(i)$ 和 $g_s(i)$；然后，用户 U_i 可以利用它在初始化阶段保留的秘密私钥 $h_s(i)$ 以及 $w_s(i)$ 和 $g_s(i)$ 值计算出当前的组密钥更新消息 $\mathrm{RK}_s=(w_s(i)-h_s(i))/g_s(i)$；随后，它利用单向函数 H 计算其他丢失的组密钥更新消息 $\mathrm{RK}_i=H^{s-i}(\mathrm{RK}_s)(r<i<s)$；最后，用户 $U_i\in G_i$ 可按下式计算当前以及先前丢失的组密钥 $\{\mathrm{TEK}_i | r<i\leqslant s\}$：

$$\mathrm{TEK}_i = f(H^i(K_0^F),\mathrm{RK}_i),\quad r<i\leqslant s。$$

因此，对在会话 r 收到 $\{B_r | 1\leqslant r<m\}$ 的用户 $U_i\in G_r$ 而言，尽管它在随后的会话 s 前一直没有收到其他 $\{B_i | r<i\leqslant s-1, r<s\leqslant m\}$，但该用户能够利用会话 s 收到的 $B_l(r\leqslant l\leqslant s)$ 恢复先前所有丢失的组密钥 $\{K_l | r\leqslant l\leqslant s\}$，即性质 4.3.5 成立：

$$H(\boldsymbol{K}_r,\boldsymbol{K}_{r+1},\cdots,\boldsymbol{K}_s \mid \boldsymbol{B}_r,\boldsymbol{B}_s,\boldsymbol{S}_i) = 0。$$

定理 4.4.2 自愈的组密钥管理协议 S-GKDS 能够保证组密钥的前向隐私性和后向隐私性。

证明：依据定义 4.3.6，我们依次讨论 $\boldsymbol{D}_S(\boldsymbol{U},t,m)$ 的前向隐私性和后向隐私性。

后向隐私性：组密钥后向隐私性的实现完全依赖于 S-GKDS 协议的前向哈希链 K^F。当一个用户在会话 s 加入到一个活动组时，GC 将与会话 s 对应的哈希链上的种子值 $H^s(K_0^F)$ 预先通过安全的信道分发给这个新的用户。显然，对在会话 s 加入活动组的用户而言，哈希函数的单向性使得它难以计算在会话 s 以前的哈希值 $\{H^j(K_0^F) | 1\leqslant j<s\}$，进而不可能利用 $\mathrm{TEK}_j=f(H^j(K_0^F),\mathrm{RK}_j)$ 计算出组密钥 $\{\mathrm{TEK}_j | 1\leqslant j<s\}$；而对会话 s 以后(含会话 s)的哈希链，组用户则能使用预分配的种子值 $H^s(K_0^F)$ 来计算在 $j_1\leqslant j\leqslant m$ 范围的哈希序列 $H^j(K_0^F)=H^{j-s}(H^s(K_0^F))$；随后，一旦用户接收到组密钥更新消息 RK_j，它便能计算在会话 j 的组密钥 $\mathrm{TEK}_j=f(H^j(K_0^F),\mathrm{RK}_j)$。故前向哈希链 K^F 的单向性使得 $H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_s | \boldsymbol{K}_{s+1},\boldsymbol{K}_{s+2},\cdots,\boldsymbol{K}_m)=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_s)$ 成立。因此，F 中任意组用户之间的密谋也无法获得组通信密钥 $\{K_j | 1\leqslant j<s\}$，即：

$$H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_s \mid \boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_m,\{\boldsymbol{S}_i\}_{U_i\in F},\boldsymbol{K}_{s+1},\boldsymbol{K}_{s+2},\cdots,\boldsymbol{K}_m) = H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_s)。$$

前向隐私性：在 S-GKDS 协议中，组密钥的前向隐私性是通过两种机制来保证的。首先，组密钥的广播更新机制提供了一种有效的前向隐私性。假定 $B\subseteq R_r\cup R_{r-1}\cup\cdots\cup R_1(|B|\leqslant t)$ 表示在会话 r 前(包括 r)被 GC 撤销的组用户集。由于所有被撤销的用户 $U_r\in B$ 均包含在组密钥更新消息 RefreshKey(或 B_j)的 $\{R\}$ 子项中；依据密钥更新消息中多项式 $g_j(x)$ 构造方式可知，等式 $\{g_j(r)=0 | \forall U_r\in R\}$ 和 $w_j(r)=h_j(r)$ 成立；因此，任何用户 $U_r\in$

B 均无法利用广播的组密钥更新消息 B_j 及其先前保存的秘密 $h_j(r)$ 去恢复当前的组密钥更新消息 RK_j，即用户无法利用公式(4.4.12)$RK_j=(w_j(i)-h_j(i))/g_j(i)$计算。

考察 B 中所有用户的同谋：其一，对 B 中所有用户 $U_b\in B$，由于 $|B|\leqslant t$，他们不可能利用这些私钥$\{h_j(b)|r\leqslant j\leqslant m,U_b\in B\}$去构建度为 t 的多项式$\{h_j(x)|r\leqslant j\leqslant m\}$。其二，对活动组的第 $j\in[r,m]$次会话，尽管所有这些用户通过密谋知道$\{h_j(r_i)|r_i\in R,j\in[1,m]\}$和 $w_j(x)$，然而我们可以随机地选取 RK'_j，并构造 $h'_j(x)$：

$$h'_j(x)=g_j(x)\cdot RK_j+h_j(x)-g_j(x)\cdot RK'_j \tag{4.4.16}$$

$h'_j(x)$是合理的，因为它能保证如下等式的成立：

$$w_j(x)=g_j(x)\cdot RK'_j+h'_j(x) \tag{4.4.17}$$

$$h'_j(r_i)=g_j(r_i)\cdot RK_j+h_j(r_i)-g_j(r_i)\cdot RK'_j=h_j(r_i) \tag{4.4.18}$$

这表明任何一个随机产生的数 RK'_j 都可能是 B 中所有被撤销用户密谋所得的组密钥，因此：

$$H(\mathbf{RK}_j\mid \mathbf{B}_j,\mathbf{B}_{j-1},\cdots,\mathbf{B}_1,\{\mathbf{S}_i\}_{U_i\in \mathbf{R}})=H(\mathbf{RK}_j),\quad r\leqslant j\leqslant m \tag{4.4.19}$$

进而，B 中用户不可能以计算的方式获取组密钥 $TEK_j=f(H^j(K_o^F),RK_j)$，其中 $r\leqslant j\leqslant m$，即

$$H(\mathbf{K}_r,\mathbf{K}_{r+1},\cdots,\mathbf{K}_m\mid \mathbf{B}_1,\mathbf{B}_2,\cdots,\mathbf{B}_m,\{\mathbf{S}_i\}_{U_i\in \mathbf{R}})=H(\mathbf{K}_r,\mathbf{K}_{r+1},\cdots,\mathbf{K}_m)。$$

另一方面，密钥更新消息的单向链($RK_1=H(RK_2)=\cdots=H^{m-1}(RK_m)$)也为协议提供了一种有效的前向隐私性。对通信组的第 j 次密钥更新，活动的组用户 U_i 可以利用公式(4.4.12)计算出 $RK_j(=H^{m-j}(RK_m))$。然而，哈希链的单向性使得它难以计算在会话 j 以后的组密钥更新序列$\{H^{m-i}(RK_m)|j<i\leqslant m\}$，从而它不可能利用式(4.4.13)计算出后续的组密钥$\{TEK_i|j<i\leqslant m\}$；另外，组用户能够使用 RK_j 来计算会话 j 以前(包括 j)的哈希序列值 $H^{m-i}(RK_1)=H^{j-i}(H^{m-j}(RK_m))$；进而能够计算在会话 j 的组密钥 $TEK_j=f(H^j(K_o^F),RK_j)$。故密钥更新消息哈希链的单向性使得 $H(\mathbf{RK}_{s+1},\mathbf{RK}_{s+2},\cdots,\mathbf{RK}_m|\mathbf{RK}_1,\mathbf{RK}_2,\cdots,\mathbf{RK}_s)=H(\mathbf{RK}_{s+1},\mathbf{RK}_{s+2},\cdots,\mathbf{RK}_m)$成立；另外，由于组通信密钥 $TEK_j=f(H^j(K_o^F),RK_j)$，故有$H(\mathbf{K}_{s+1},\mathbf{K}_{s+2},\cdots,\mathbf{K}_m|\mathbf{K}_1,\mathbf{K}_2,\cdots,\mathbf{K}_s)=H(\mathbf{K}_{s+1},\mathbf{K}_{s+2},\cdots,\mathbf{K}_m)$成立。因此，$B$ 中任何组用户之间的密谋也无法获得组密钥$\{K_j|r\leqslant j\leqslant m\}$，即如下结论成立：

$$H(\mathbf{K}_r,\mathbf{K}_{r+1},\cdots,\mathbf{K}_m\mid \mathbf{B}_1,\mathbf{B}_2,\cdots,\mathbf{B}_m,\{\mathbf{S}_i\}_{U_i\in \mathbf{R}},\mathbf{K}_1,\mathbf{K}_2,\cdots,\mathbf{K}_{r-1})=H(\mathbf{K}_r,\mathbf{K}_{r+1},\cdots,\mathbf{K}_m)。$$

综上所述，S-GKDS 协议的组密钥分发机制能够满足前向/后向隐私性需求。

4.4.5 性能分析

除协议的安全性以外，考虑到某些实际应用中组用户节点较低的计算能力(如无线应用)和较高的信道误码率，性能也是一个很重要的因素。考虑到协议的动态特性，我们需要对组密钥更新消息 RK 的性能进行评价，主要是量化组用户更新组密钥时的通信开销和计算开销。

我们首先使用 Markov 链来得到用户密钥更新消息缓冲区状态的稳态分布；然后重点讨论 S-GKDS 协议的计算和通信开销；最后与其他相关的研究工作进行了对比分析。

4.4.5.1 稳态 Markov 状态分布

如图 4.4.6 所示，我们用 Markov 链对一个组用户的消息缓冲区的状态分布进行了建模。我们假设：①每个 RefreshKey 消息包的丢失是随机发生且相互独立的事件；②无效的 RefreshKey 消息包是随机发生且相互独立的事件；③如果一个用户的组密钥更新消息缓冲区用完，每个用户都会在间隔 T_{refresh} 内完成 RequestGroupKey 消息操作。

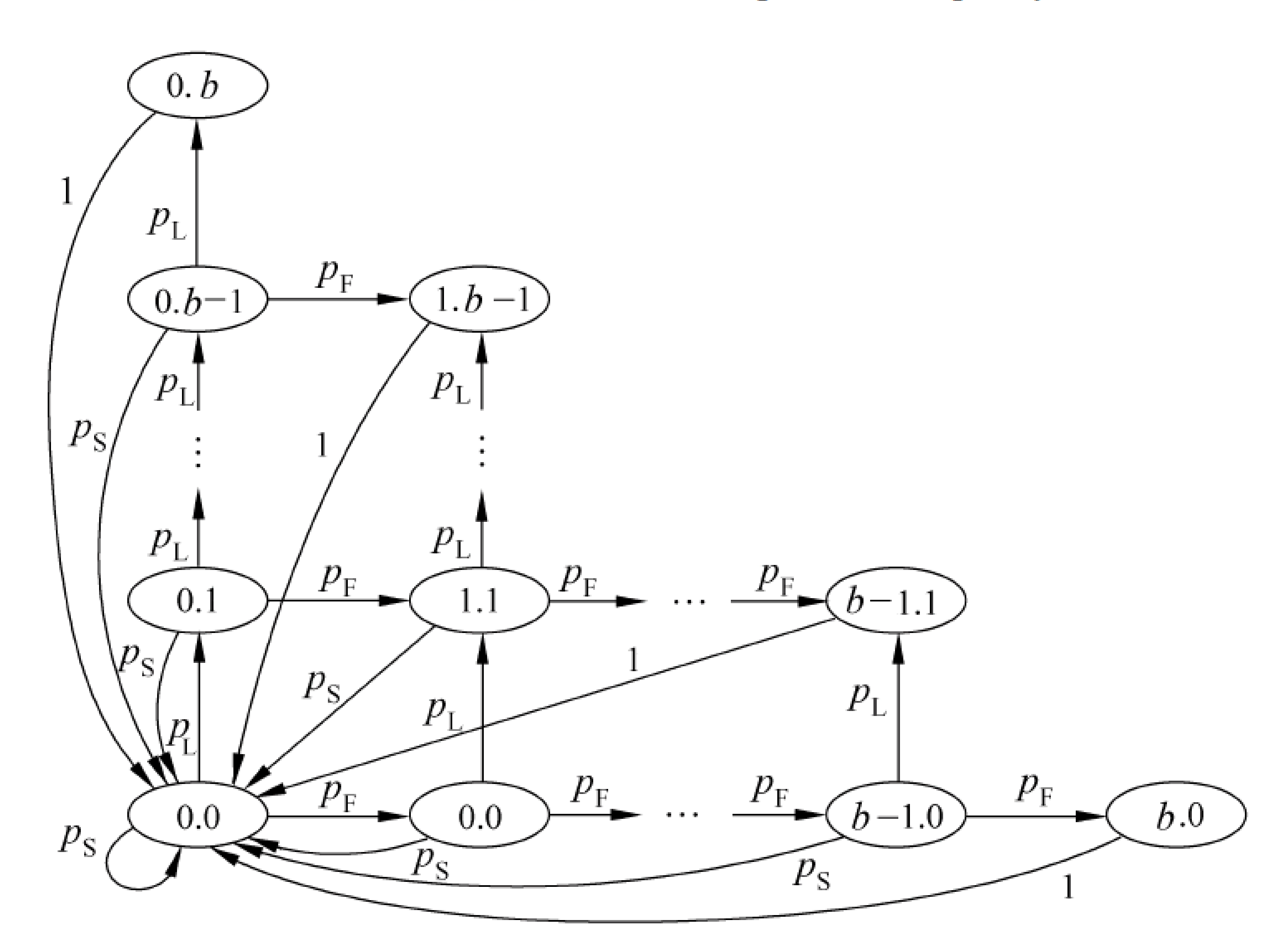


图 4.4.6 组用户消息缓冲区的状态变迁图

不失一般性，我们假定所有的消息包（有效或无效）在传输过程中具有相同的包丢失率，令 $p_{\mathrm{L}}=\Pr\{\text{RK 消息丢失}\}$。该假定是合理的，因为通信链路在传输数据包时，并不区分不同包的数据类型。p_{L} 反映了通信链路的信道状况，即一个高的 p_{L} 表明该链路具有较高的包丢失率和错误率。此外，令 $p_{\mathrm{S}}=\Pr\{\text{RK 消息包认证有效}\mid\text{RK 消息被接收}\}$，$p_{\mathrm{F}}=\Pr\{\text{RK 消息包认证无效}\mid\text{RK 消息被接收}\}$。$p_{\mathrm{L}}$，$p_{\mathrm{S}}$ 和 p_{F} 三者之间存在如下关系：

$$p_{\mathrm{L}}+p_{\mathrm{F}}+p_{\mathrm{S}}=1 \tag{4.4.20}$$

图 4.4.6 中的状态 $S_{i,j}$ 表示组用户已经有 i 个密钥更新周期未收到 RefreshKey 消息包，或者已收到 j 个无效的消息包。相应地，处于该状态下的用户，其组密钥更新消息缓冲区 kb 中将出现 $i+j$ 个空槽。状态变迁在如下 3 种情况下被触发：RefreshKey 消息包丢失、接收到无效的 RefreshKey 消息包和接收到有效的 RefreshKey 消息包。

令 $P(i,j)$ 表示状态 $S_{i,j}$ 处于稳定状态时的概率，则有

$$P(i,j)=\begin{cases} p_{\mathrm{F}}\cdot P(i,j-1), & i=0,j>0 \\ p_{\mathrm{L}}\cdot P(i-1,j), & i>0,j=0 \\ p_{\mathrm{F}}\cdot P(i,j-1)+p_{\mathrm{L}}\cdot P(i-1,j), & i>0,j>0 \\ pt/(p_{\mathrm{F}}+p_{\mathrm{L}}), & i=j=0 \end{cases} \tag{4.4.21}$$

其中，

$$pt=\sum_{i+j=b}P(i,j)+p_{\mathrm{S}}\cdot\sum_{i+j<b,(i,j)\neq(0,0)}P(i,j) \tag{4.4.22}$$

实际上，我们可以用另外一种方法来简化如上稳态概率的求解。对组用户而言，RefreshKey 消息包丢失（发生概率是 p_L）和收到无效的 RefreshKey 消息包（发生概率是 p_F）这两种事件是等价的，因为它们对变迁的触发具有对称性。若我们令状态 S_k 表示用户的组密钥更新消息缓冲区 kb 中有 i 个空槽，则利用 Markov 链的状态对称简化技术，图 4.4.6 的状态变迁图可以进一步简化为如图 4.4.7 所示的等价状态变迁图。

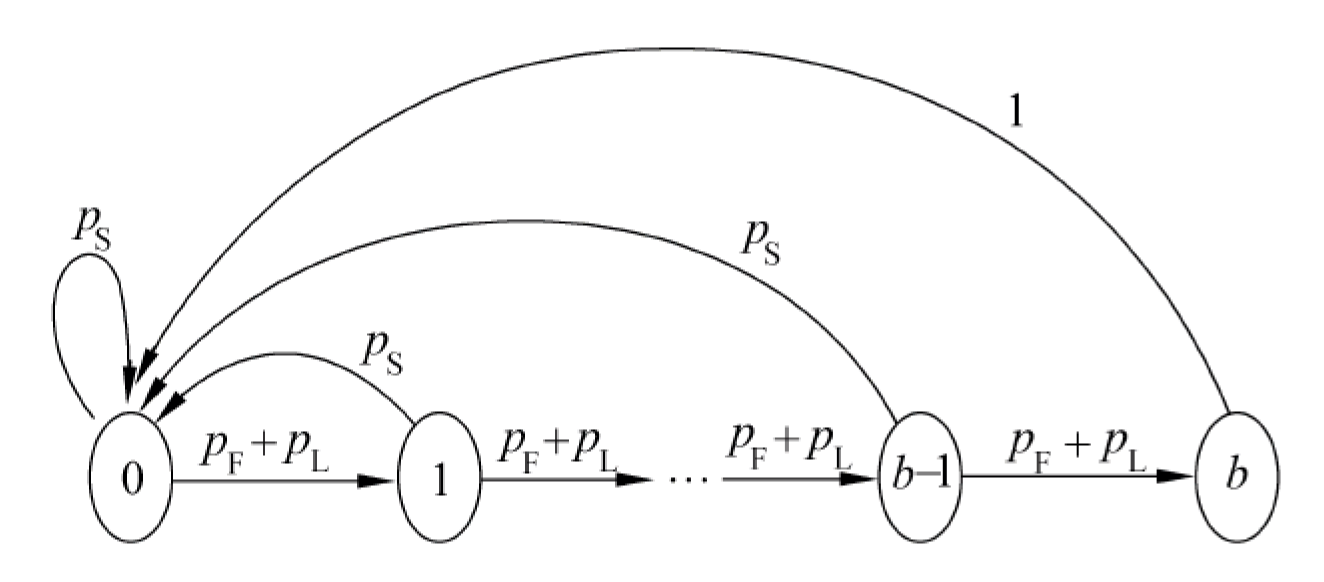


图 4.4.7　简化的等价状态变迁图

令 $P(k)$ 为用户的消息缓冲区 kb 中恰好有 k 个空槽时的稳态概率，则有

$$\sum_{k=0}^{b} P(k) = 1 \tag{4.4.23}$$

根据全局状态平衡方程，有

$$\begin{cases} P(i)\cdot(p_S+p_F+p_L) = P(i-1)\cdot(p_F+p_L), & i=1,2,\cdots,b \\ P(0)\cdot(p_F+p_L) = P(t) + \sum_{i=1}^{b-1} P(i)\cdot p_S \end{cases} \tag{4.4.24}$$

则稳态分布 $P(k)$ 由下式得到：

$$\begin{cases} P(0) = (1-p_L)/(1-p_L^{b+1}) \\ P(k) = P(0)\cdot p_L^k, & k=1,2,\cdots,b \end{cases} \tag{4.4.25}$$

4.4.5.2 通信开销

基于用户的状态变迁模型，我们可以定量地分析 GC 和用户之间的通信开销。令 C_{init} 和 C_{refresh} 分别为传输 InitGroupKey 和 RefreshKey 消息的通信开销，令比率 $\alpha = C_{\text{init}}/C_{\text{refresh}}$，则很明显 $\alpha > 1$，因为协议在传输 InitGroupKey 消息有时需要比 RefreshKey 消息有更多的带宽资源。

在以下两种情况下，GC 必须传输 InitGroupKey 消息给组用户：①当一个新用户加入到活动的组会话时，GC 需要把当前的组密钥更新参数通过 InitGroupKey 消息发送给该用户；②当组用户有多于 b 个 RefreshKey 消息丢失或认证失败时，组用户需向 GC 发送 RequestKey 消息以明确请求当前的组密钥更新消息。GC 在收到该请求后，则需要发送一个包括当前密钥更新参数的 InitGroupKey 消息给该用户。

在这两种情况下，GC 需要发送 InitGroupKey 消息对请求的用户进行初始化。此外，GC 还需要周期性地广播 RefreshKey 消息，因此一个用户通信开销的期望值 C_{Comm} 是

$$E[C_{\text{Comm}}] = C_{\text{init}}\cdot[P(t)+p_j] + C_{\text{refresh}}\cdot\sum_{k=0}^{b-1} P(k) \tag{4.4.26}$$

其中，p_j 为组用户加入组会话的概率。为方便分析，我们利用发送 RefreshKey 消息的通信开销 C_{init} 来规范化 C_{Comm}，即：

$$NC_{Comm} = \alpha \cdot [(p_L + p_F)^b \cdot P(0) + p_j] + \sum_{k=0}^{b-1} p_L^k \cdot P(0) \tag{4.4.27}$$

为了分析组密钥更新消息的动态特性，我们忽略组用户加入的通信开销，因此，式(4.4.27)可进一步被简化为

$$NC_{Comm} = \alpha \cdot [(p_L + p_F)^b \cdot P(0)] + \sum_{k=0}^{b-1} p_L^k \cdot P(0) \tag{4.4.28}$$

如果 C_{Comm} 接近于 1，则说明 RefreshKey 消息能够提供一种良好的组密钥更新机制。如果 C_{Comm} 接近 α，则说明该协议效率较低。图 4.4.8 描述了在 $p_L=0.1\sim0.5$ 和 $\alpha=10$ 的情况下，C_{Comm} 和密钥缓冲区长度 b 之间的函数关系。选择 $\alpha=10$ 意味着传输和处理 InitGroupKey 的开销比传输 RefreshKey 的开销要高，因为 InitGroupKey 包比 RefreshKey 包要大。结果表明，每个用户的密钥缓冲区长度决定了它的通信开销，较小的 b 值将导致一个高的通信开销，而较大的 b 值则会显著减少通信开销。

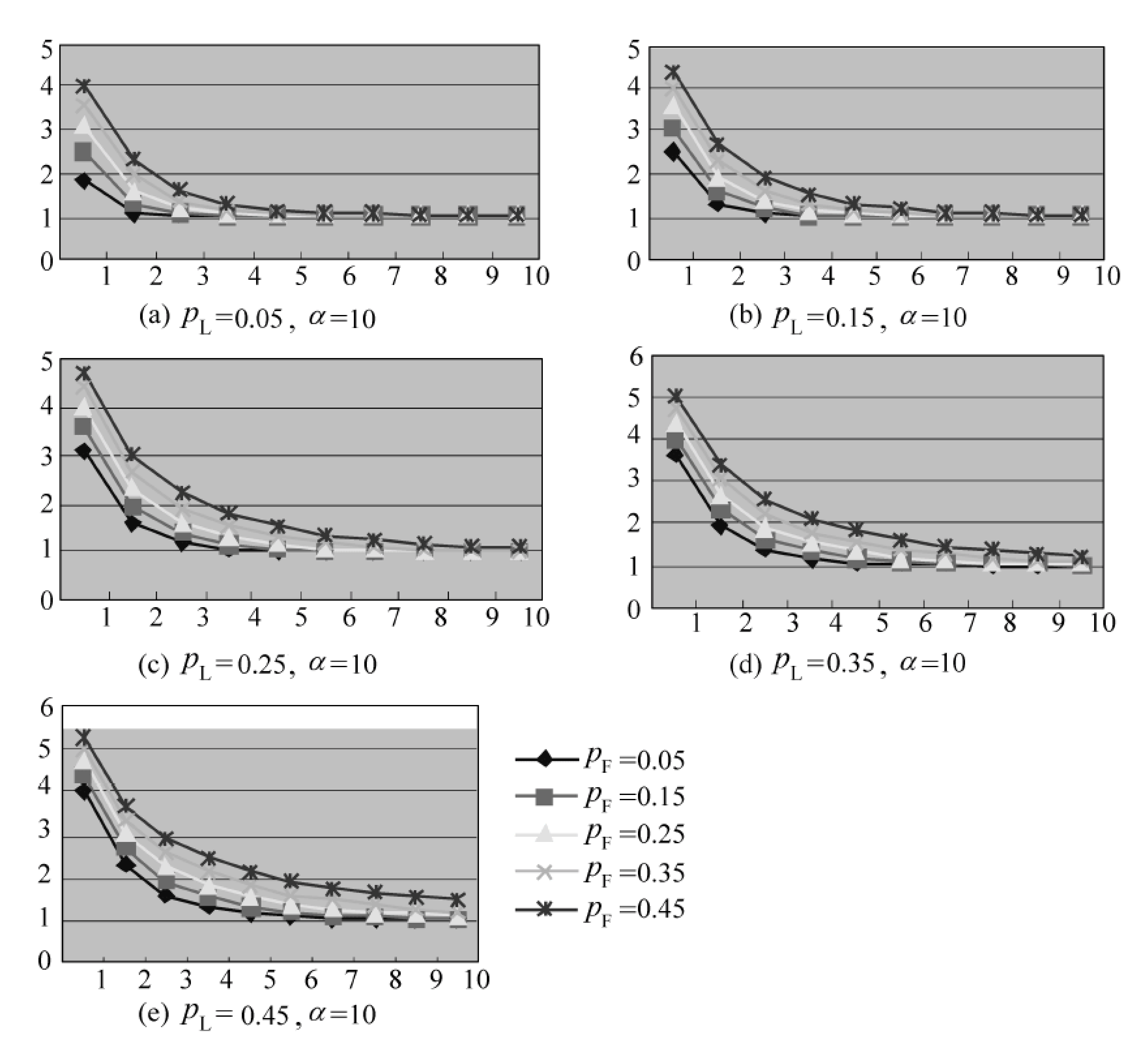


图 4.4.8　规范化通信开销 NC_{Comm} 和组密钥更新消息缓冲区长度 b 的关系

($p_F=0.05\sim0.45, \alpha=10$)

图 4.4.8 表明，即使在包丢失率较大的情况下($p_L\geqslant0.3$)，该协议的通信开销也是有效的，因为如果 $b\geqslant10$，使用 C_{init} 规范化后的通信开销 NC_{Comm} 接近于 1。

4.4.5.3 计算开销

GC 通常是一个高性能的服务器，能够处理复杂的计算任务。因此，我们仅重点讨论组用户节点的计算复杂度。由 S-GKDS 协议的机制可知，用户的计算开销主要是在处理 RefreshKey 或 InitGroupKey 消息所耗费的哈希运算。令 N_h 表示组用户在处理 RefreshKey 或 InitGroupKey 消息时所进行的哈希计算次数。若节点密钥更新消息缓冲区的空槽数 $k<b$，则需要进行的哈希计算次数 N_h 为

$$N_h=\begin{cases}2, & \text{若组密钥更新消息 RefreshKey 丢失}\\ k+3, & \text{若组密钥更新消息 RefreshKey 认证无效}\\ k+3, & \text{若组密钥更新消息 RefreshKey 认证有效}\end{cases} \tag{4.4.29}$$

组用户可能因 RK 的认证无效或者组密钥更新消息 RefreshKey 的丢失，而导致其组密钥消息缓冲区里所有 b 个槽全为空，即 $k=b$；这时，组用户将向 GC 发送 RequestKey 消息以得到新的组密钥更新消息。而该用户在接收到新的初始化消息 InitGroupKey 时，需要进行额外的 $b+1$ 次哈希计算，即：

$$N_h=\begin{cases}b+3, & \text{若组密钥更新消息 RefreshKey 丢失}\\ 2b+4, & \text{若组密钥更新消息 RefreshKey 认证无效}\\ b+3, & \text{若组密钥更新消息 RefreshKey 认证有效}\end{cases} \tag{4.4.30}$$

相应地，若组用户的密钥更新消息缓冲区有 k 个空槽，则相应的条件期望值 $E[N_h|$缓冲区有 k 个空槽$]$为

$$\begin{aligned}E[N_h\mid k\text{ 个空槽}]&=\begin{cases}2\cdot p_{\mathrm{L}}+(k+3)\cdot(p_{\mathrm{S}}+p_{\mathrm{F}}), & k<b\\ (b+1)+(b+1)\cdot p_{\mathrm{F}}, & k=b\end{cases}\\ &=\begin{cases}(k+3)\cdot(1-p_{\mathrm{L}})+2\cdot p_{\mathrm{L}}, & k<b\\ (b+3)\cdot(1+p_{\mathrm{F}})+(b+1)\cdot p_{\mathrm{F}}, & k=b\end{cases}\end{aligned} \tag{4.4.31}$$

最后，根据用户的稳定状态分布概率 $P(k)=P(0)\cdot p_{\mathrm{L}}^k, k=1,2,\cdots,b$，我们可以进一步导出 N_h 的期望值 $E[N_h]$为

$$\begin{aligned}E[N_h]&=\sum_{k=0}^{b}E[N_h\mid \text{缓冲区 kb 中有 } k \text{ 个空槽}]\cdot P(k)\\ &=\{(b+3)\cdot(1+p_{\mathrm{F}})+(b+1)\cdot p_{\mathrm{F}}\}\cdot P(0)\cdot(p_{\mathrm{L}}+p_{\mathrm{F}})^b\\ &\quad+\sum_{k=0}^{b-1}\{(k+3)\cdot(1-p_{\mathrm{L}})+2p_{\mathrm{L}}\}\cdot P(0)\cdot(p_{\mathrm{L}}+p_{\mathrm{F}})^k\end{aligned} \tag{4.4.32}$$

这里，$E[N_h]$可用来近似表示组用户的计算开销 C_{Comp}，即

$$C_{\mathrm{Comp}}=E[N_h] \tag{4.4.33}$$

图 4.4.9 描述了计算开销 C_{Comp} 和 p_{L} 的函数关系，其中 p_{L} 从 0.05 变化到 0.45。图 4.4.10 描述了计算开销 C_{Comp} 和组密钥更新长度 b 的函数关系，其中的 p_{L} 从 0.05 变化到 0.45，p_{F} 则从 0.05 变化到 0.35。图 4.4.11 则描述了计算开销 C_{Comp} 和 p_{F} 的函数关系，其中 p_{F} 从 0.05 变化到 0.45。

图 4.4.10 的分析结果表明，每个组用户节点的计算开销是很低的，因为即使在非常恶劣的通信环境下，如 $p_{\mathrm{L}}=0.5$，每个用户对处理 RefreshKey 消息(操作 RK)仅需要计算少于两次哈希函数。

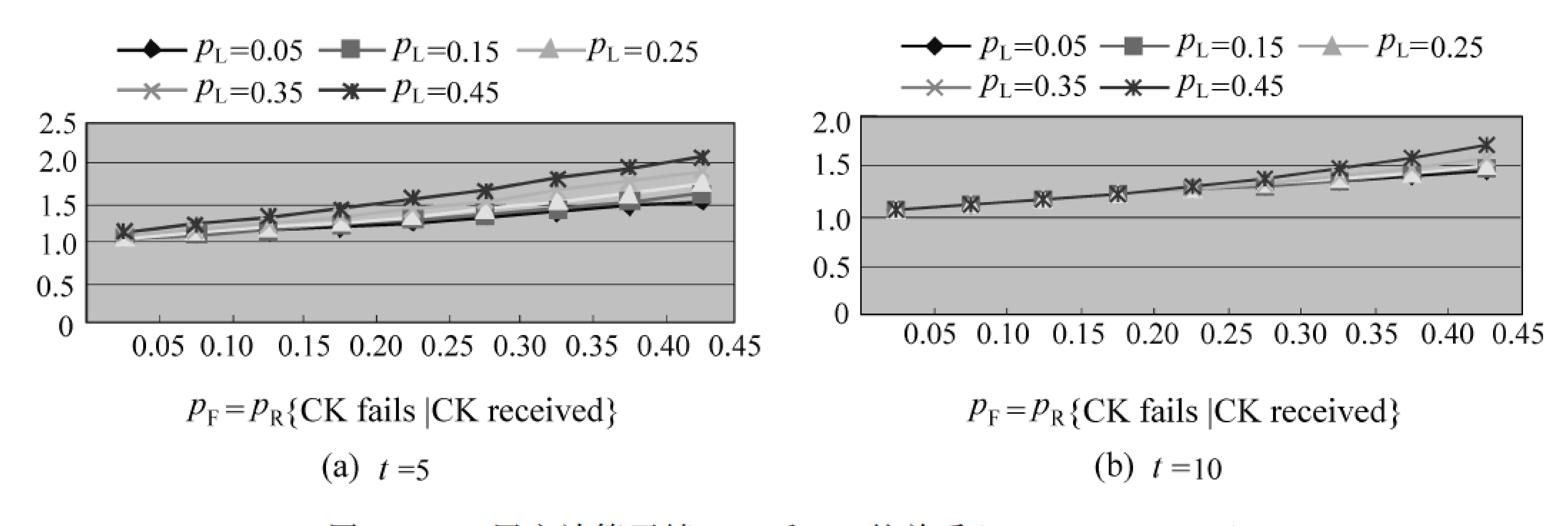


图 4.4.9　用户计算开销 C_{Comp} 和 p_L 的关系（p_L＝0.05～0.45）

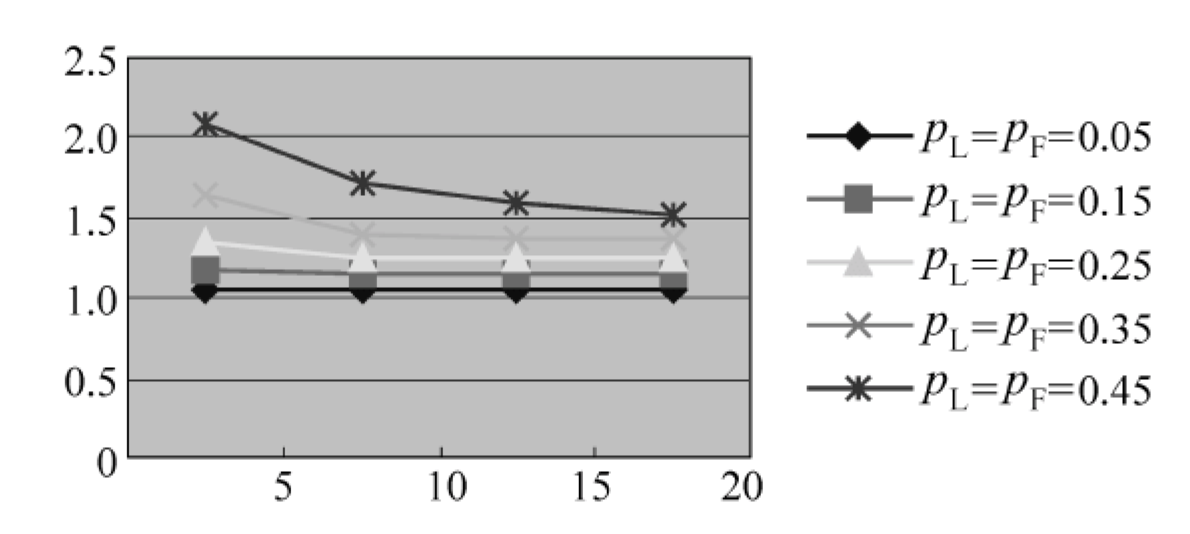


图 4.4.10　组用户的计算开销与组密钥更新长度 b 的关系（m＝500）

综上所述，图 4.4.9～图 4.4.11 的模拟分析结果表明，为了保证较低的通信和计算开销，理想的组密钥更新长度应满足 $b \geqslant 10$。在这种情况下，规格化的通信和计算开销位于 1～1.5 之间；这说明，即使在信道较高的丢包率和错误率下，S-GKDS 协议在通信和计算开销上是仍然是有效的。

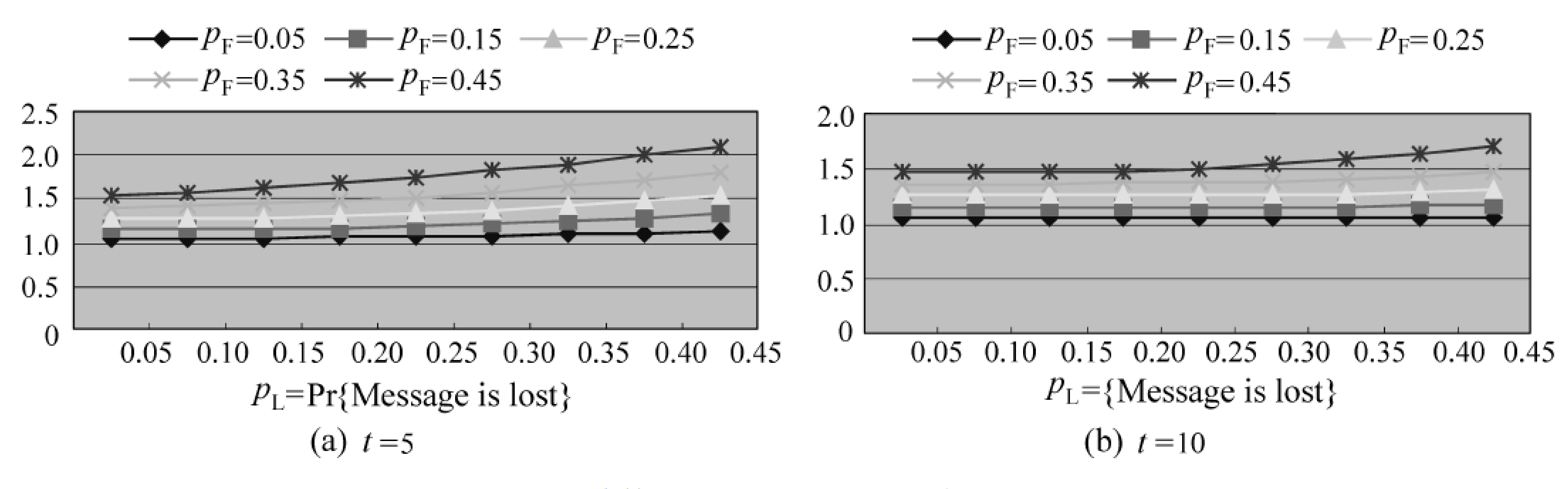


图 4.4.11　组用户计算开销 C_{Comp} 和 p_F 的关系（p_F＝0.05～0.45）

4.4.5.4　与其他类似协议的对比分析

表 4.4.1 对 S-GKDS 协议、Stadden 的自愈协议[10]以及 Liu-Ning[57]的自愈协议的性能进行了对比分析。

存储开销：在初始化阶段，每个组用户 U_i 需要存储自己的身份标识 i 和掩码多项式 $\{h_j(x)\}_{j=1,2,\cdots,z} \in F_q[x]$在点 i 处的值$\{h_1(i),h_2(i),\cdots,h_z(i)\}$。因此，对每个用户 U_i 而言，其存储复杂度为 $O(m\log q)$。因此，就存储开销而言，S-GKDS 协议和 Liu-Ning 的自愈协议

表 4.4.1 性能对比

	通信开销(组播)	通信开销(单播)	存储开销
S-GKDS	$O(t\log q)$	$O(m\log q)$	$O(m\log q)$
Stadden[10]	$O((mt^2+mt)\log q)$	$O(m^2\log q)$	$O(m^2\log q)$
Liu-Ning[57]	$O((mt+m+t)\log q)$	$O(m\log q)$	$O(m\log q)$

基本一样,都为 $O(m\log q)$;而 Stadden 的自愈协议则为 $O(m^2\log q)$,其存储优化的效果是显著的。

通信开销:广播消息 $\boldsymbol{B}_j$ 包括在第 j 次会话中由 GC 所撤销的组用户标识集和一个 t 维多项式 $w_j(x)$。因此,S-GKDS 的通信开销是 $O(t\log q)$。相应地,Stadden 的组密钥分发协议则为 $O((mt^2+mt)\log q)$,Liu-Ning 的组密钥分发协议则为 $O((mt+m+t)\log q)$。显然,S-GKDS 协议使得 GC 和组用户之间的广播通信开销得到了显著的优化,因为广播消息包的大小被减少到 $O(t\log q)$。特别地,由于 S-GKDS 协议的通信开销独立于组通信的最大会话次数 m,只与撤销的最大用户数 t 相关,这使得当 m 较大时,协议的优化效果将更为显著。

4.5 基于时限用户撤销机制的自愈组密钥分发协议

在 B-GKDS 和 S-GKDS 协议中,组用户的撤销机制是一种显式的撤销机制,完全通过组密钥更新消息 RefreshKey 来实现。从 S-GKDS 协议的性能分析可知,这种机制具有较高的效率,但也存在一定的局限性,即被撤销的最大用户数受制于门限值 t。由于S-GKDS 协议的通信开销是 $O(t\log q)$,增大门限值 t 将显著增大消息 RefreshKey 的包长度;另一方面,S-GKDS 的协议机制也不利于动态到调整门限值 t。

在自愈的组密钥分发协议 S-GKDS 的基础上,提出了一种基于时限用户撤销机制的组密钥分发协议[70](self-healing group key distribution scheme with time-limited user revocation,S-GKDS-TL)。该协议是 S-GKDS 协议的扩展,它几乎完整地保留了 S-GKDS 协议的所有基本特性,如安全性、自愈性、自适应性和高性能,并克服了 S-GKDS 协议中要求被撤销的最大用户数不超过门限值 t 这一限制。

S-GKDS-TL 协议的实现可以依赖两种模式:基于双向哈希链(dual directional hash chains,DDHC)的方法和基于单向哈希树(hash binary tree,HBT)的方法。DDHC 和 HBT 本身并不是一种组密钥分发机制,但是它是时限用户撤销算法的基础。HBT 方法比 DDHC 方法具有更强的安全性。

与 S-GKDS 协议相比,S-GKDS-TL 协议具有更优的动态性能,能够更好地适应中大规模的组通信环境中用户频繁地参与或退出会话。因为时限用户撤销机制使得组用户能够以一种隐式方式退出组会话,则这种机制具有:①用户的退出并不需要组管理中心 GC 的直接介入;②协议对撤销用户的总数没有限制;③用户撤销算法具有较小的计算和通信开销。时限用户撤销机制的引入使得 S-GKDS-TL 协议提供了两种高效的组用户管理机制:其一是通过组密钥广播更新消息实现的显式用户撤销;其二是通过时限用户撤销机制实现

的隐式用户撤销。这两种撤销机制的存在，使得协议在组用户频繁参与或退出组会话时能够提供更好的自适应性。

S-GKDS-TL 协议具有和 S-GKDS 协议一样的安全性。我们利用信息熵概念形式化地定义和描述了 S-GKDS-TL 协议的信息熵模型，并证明了 S-GKDS-TL 协议就信息论范畴而言是无条件安全的。

S-GKDS-TL 协议继承了 S-GKDS 协议的所有基本特性，并且时限用户撤销机制引入并未给协议本身带来显著的计算和通信负载，因为这是一种轻量级的算法。因此，与 S-GKDS协议相似，S-GKDS-TL 协议同样可以应用于无线网络、移动网络（NEMO 网）和无线传感器网络这些具体的组通信环境。

此外，S-GKDS-TL 协议能够抵御多个用户的同谋攻击，因此，S-GKDS-TL 协议具有比 S-GKDS 协议更好的安全性。同样，利用 S-GKDS-TL 协议的信息熵模型，我们证明了 S-GKDS-TL 协议在信息论范畴内是无条件安全的。

4.5.1　S-GKDS-TL 协议的信息熵模型

这里，我们给出 S-GKDS-TL 协议的信息熵模型。S-GKDS-TL 协议完全是 S-GKDS 协议的扩展，它在 S-GKDS 协议的基础上，引入了组密钥的时限访问机制。因此，在如下 S-GKDS协议的信息熵模型的定义中，我们主要引入了时限组用户撤销机制的形式化定义，其他与 B-GKDS 协议的信息熵模型一致。

定义 4.5.1　假定 $\boldsymbol{U}=\{U_1,U_2,\cdots,U_n\}$是所有可能的组通信用户的集合，$m$ 是组通信系统的最大会话次数，t 是 GC 所能主动撤销的最大用户数，则 $\boldsymbol{D}_{\text{S-TL}}(\boldsymbol{U},t,m)$是一个基于时限用户撤销机制的自愈组密钥分发模型，若如下条件能够满足：

（1）对组用户 $U_i\in G_i$ 而言，组密钥 K_j 完全可以由 B_j 和它的私钥 S_i 决定，即

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j,\boldsymbol{S}_i)=0 \tag{4.5.1}$$

（2）对任何用户子集 $B\subseteq\boldsymbol{U}$，$|B|\leqslant t$，集合 B 中的用户不可能获得用户 $U_k\notin B$ 的用户私钥 S_k，即

$$H(\boldsymbol{K}_j,\boldsymbol{S}_k \mid \boldsymbol{B}_j,\boldsymbol{B}_{j-1},\cdots,\boldsymbol{B}_1,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{B}})=H(\boldsymbol{K}_j,\boldsymbol{S}_k) \tag{4.5.2}$$

（3）不可能单独从 GC 广播的密钥更新信息或组用户私钥获得组密钥，即

$$H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m \mid \boldsymbol{B}_1,\boldsymbol{B}_2,\cdots,\boldsymbol{B}_m)=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m) \tag{4.5.3}$$

$$H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m \mid \boldsymbol{S}_1,\boldsymbol{S}_2,\cdots,\boldsymbol{S}_n)=H(\boldsymbol{K}_1,\boldsymbol{K}_2,\cdots,\boldsymbol{K}_m) \tag{4.5.4}$$

（4）$\boldsymbol{D}_{\text{S-TL}}(\boldsymbol{U},t,m)$能够同时安全撤销最多 t 个用户的能力：设每次会话 j 被撤销的组用户集是 $R=R_j\cup R_{j-1}\cup\cdots\cup R_1$，$|R|\leqslant t$，则 R 中的组用户不可能利用 GC 广播的组密钥更新信息 B_j 去恢复出当前的组通信密钥 K_j，即

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j,\boldsymbol{B}_{j-1},\cdots,\boldsymbol{B}_1,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{R}})=H(\boldsymbol{K}_j) \tag{4.5.5}$$

（5）对用户 $U_i\in G_r$，若其在会话 r 收到密钥更新消息$\{B_r|1\leqslant r<m\}$，但在会话 s 前一直没有被撤销（$r<s\leqslant m$），则该用户可利用会话 s 收到的密钥更新消息$\{B_l|r\leqslant l\leqslant s\}$恢复所有的组通信密钥$\{K_l|r\leqslant l\leqslant s\}$，即

$$H(\boldsymbol{K}_r,\boldsymbol{K}_{r+1},\cdots,\boldsymbol{K}_s \mid \boldsymbol{B}_r,\boldsymbol{B}_s,\boldsymbol{S}_i)=0 \tag{4.5.6}$$

（6）（时限访问特性）对活动周期为$[r,s]$的用户 $U_i\in G_r$ 而言，其在会话 r 加入组通信，

则 U_i 仅能获得$[r,s]$之间的组密钥 $K_j(r\leqslant j\leqslant s)$；而无法获得$[1,r-1]$和$[s+1,m]$之间的组密钥，即

$$H(\boldsymbol{K}_1,\cdots,\boldsymbol{K}_{r-1},\boldsymbol{K}_{s+1},\cdots,\boldsymbol{K}_m \mid \boldsymbol{B}_1,\cdots,\boldsymbol{B}_m,\{\boldsymbol{S}_i\}_{U_i\in \boldsymbol{G}})$$
$$=H(\boldsymbol{K}_1,\cdots,\boldsymbol{K}_{r-1},\boldsymbol{K}_{s+1},\cdots,\boldsymbol{K}_m) \tag{4.5.7}$$

定义 4.5.1 的性质(1)～性质(5)完整地继承了定义 4.4.1 的性质；而定义 4.5.1 中的性质(6)则描述了组密钥分发模型 $\boldsymbol{D}_{\text{S-TL}}(\boldsymbol{U},t,m)$的时限用户撤销机制。同样，考虑到组密钥分发协议的安全性需求，$\boldsymbol{D}_{\text{S-TL}}(\boldsymbol{U},t,m)$还应满足组密钥的前向隐私性和后向隐私性(定义 4.3.6)。

4.5.2　隐式组用户撤销机制

在 S-GKDS 协议中，组用户的撤销机制是一种显式撤销机制，它完全通过组密钥更新消息 RefreshKey 来实现。具体而言，组密钥的更新消息保证了组密钥的后向隐私性；而组密钥的前向隐私性则是通过前向哈希链 K^F 来保证的。

与 S-GKDS 类似，S-GKDS-TL 协议的隐式用户撤销机制同样依赖于哈希链的单向性；所不同的是，S-GKDS-TL 通过引入双向哈希链(dual directional hash chains，DDHC)来保证组密钥的前向和后向隐私性。

4.5.2.1　双向哈希链

双向哈希链(DDHC)由两个同等长度的单向哈希链组成：前向哈希链 K^F 和后向哈希链 K^B。每一个哈希链通过在一个秘密种子上重复应用一个单向哈希函数 H 来产生，DDHC 由以下步骤产生：①产生两个随机密钥种子值，K_0^F 和 K_0^B，前向和后向哈希函数的长度为 m；②对每个种子值重复使用相同的单向哈希函数 H 产生两个长度同为 m 的哈希链：

$$\{K_0^F,H(K_0^F),\cdots,H^i(K_0^F),\cdots,H^{m-1}(K_0^F)\},$$
$$\{K_0^B,H(K_0^B),\cdots,H^i(K_0^B),\cdots,H^{m-1}(K_0^B)\}。$$

S-GKDS-TL 通过引入了双向哈希链：前向哈希链 K^F 和后向哈希链 K^B，来分别保证组密钥的后向和前向隐私性，同时也提供了一种有效的隐式用户撤销机制。

4.5.2.2　时限用户撤销机制

时限用户的撤销机制适用于不同时间段采用不同密钥加密的安全组通信应用中。如电视节目不同时间段采用不同的密钥加密。

时限用户撤销机制是由 DDHC 实现的。令 m 是组通信系统的最大会话次数。在 S-GKDS-TL 中，所有的组用户均有明确的活动周期$[j_1,j_2]$，即该用户在会话 j_1 加入活动组，而在会话 j_2 退出活动组。另外，在活动周期$[j_1,j_2]$期间，GC 还可以利用 S-GKDS 的用户撤销方式主动、强制性地显式撤销该用户。

基于双向哈希链 DDHC 的隐式用户撤销机制如图 4.5.1 所示，这是一种时限的撤销方式。图 4.5.1 中阴影部分的哈希链表示活动周期为$[j_1,j_2]$的组用户所不能访问的哈希值。当一个组用户 U_i 在会话 j_1 加入到一个活动组时，GC 将根据用户 U_i 的活动周期$[j_1,j_2]$分配一对哈希值$(H^{j_1}(K_0^F),H^{m-j_2}(K_0^B))$给该用户：其中 $H^{j_1}(K_0^F)$对应于前向链 K^F 上的哈希

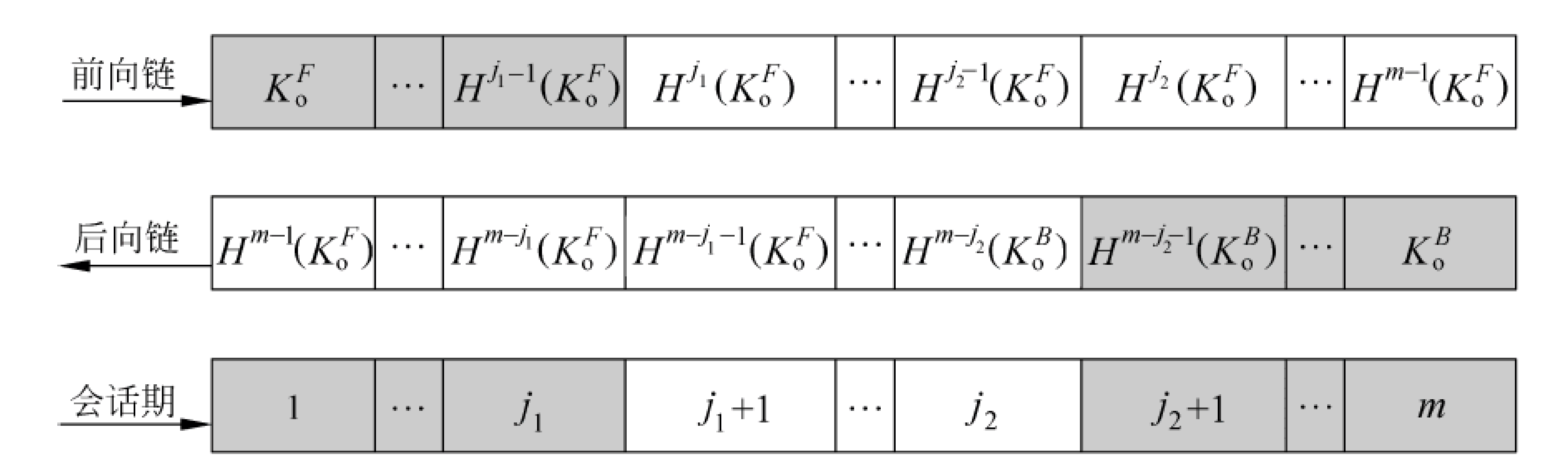


图 4.5.1　隐式用户撤销：基于双向哈希链 DDHC 的时限用户撤销

值，而 $H^{m-j_2}(K_o^B)$对应于后向链 K^B 上的哈希值。显然，这种双向哈希链使得活动周期为(j_1,j_2)的组用户只能同时访问在 $j_1\leqslant j\leqslant j_2$ 范围内的 DDHC，即$\{(H^j(K_o^F),H^{m-j}(K_o^B))|j_1\leqslant j\leqslant j_2\}$。

DDHC 的前向哈希链 K^F 有效地保证了组密钥的后向隐私性。对在会话 j_1 加入活动组的用户而言，由于哈希链的单向性，它难以计算在会话 j_1 以前的哈希值序列$\{H^j(K_o^F)|1\leqslant j<j_1\}$；而对会话 j_1 以后(含会话 j_1)的哈希值，组用户则能使用预先分配的哈希种子值 $H^{j_1}(K_o^F)$来计算前向链 K^F 中对应的哈希值序列$\{H^j(K_o^F)|j_1\leqslant j\leqslant m\}$：

$$H^j(K_o^F)=H^{j-j_1}(H^{j_1}(K_o^F))\tag{4.5.8}$$

组密钥的前向隐私性则依赖于 DDHC 的后向哈希链 K^B。对于活动周期为$[j_1,j_2]$的组用户而言，它将在会话 j_2 后离开活动组。后向哈希链 K^B 的单向性，使得它难以计算在会话 j_2 以后的哈希值序列$\{H^j(K_o^B)|j_2<j\leqslant m\}$；而对会话 j_2 以前(含会话 j_2)的哈希值，组用户则能使用预先分配的哈希值 $H^{m-j_2}(K_o^B)$来计算 K^B 中对应的哈希值序列$\{H^{m-j}(K_o^F)|1\leqslant j\leqslant j_2\}$：

$$H^{m-j}(K_o^B)=H^{j_2-j}(H^{m-j_2}(K_o^B))\tag{4.5.9}$$

在 S-GKDS-TL 协议中，对活动周期是$[1,m]$的组系统，会话 j 的组通信密钥被定义为 j，$H^j(K_o^F)$，$H^{m-j}(K_o^B)$和 RK_j 的函数：

$$\mathrm{TEK}_j=f(H^j(K_o^F),H^{m-j}(K_o^B),\mathrm{RK}_j)\tag{4.5.10}$$

其中，$1\leqslant j\leqslant m$；K_o^F 和 K_o^B 分别是前向和后向哈希链的种子值；函数 $f(\cdot)$也是一个单向函数，它能把任意长度的消息经过处理后输出为一个固定长度的值；RK_j 来源于 GC 的第 j 次组密钥更新消息，其更新处理机制与 S-GKDS 相似。

从组通信密钥的构造方式可知，双向哈希链 DDHC 能够有效地同时保证组密钥的前向和后向隐私性。双向哈希链 DDHC 的单向性使得只有同时具有哈希值$\{H^{j_1}(K_o^F),H^{j_2}(K_o^B)\}$的组用户 U_i 才能访问在(j_1,j_2)范围的组密钥 TEK。因为 U_i 能够使用预先分发的哈希值$\{H^{j_1}(K_o^F),H^{m-j_2}(K_o^B)\}$计算$\{H^j(K_o^F)|j_1\leqslant j\leqslant m\}$和$\{H^{m-j}(K_o^B)|1\leqslant j\leqslant j_2\}$；而对于范围$(j_1,j_2)$外的组会话，如 $j<j_1$ 或 $j>j_2$，组用户则不可能同时计算出 $H^j(K_o^F)$和$H^{m-j}(K_o^B)$，进而也不可能计算出在会话 j 时的 TEK。因此每个组用户只能按照它的活动周期$[j_1,j_2]$，在预先定义的活动周期内访问组密钥并参与组通信。

这是一种隐性的限时用户撤销方法，因为一旦 U_i 的活动周期过期($j>j_2$)，它将自动退出通信组而不需要 GC 的直接干涉。

4.5.3 S-GKDS-TL 组密钥分发协议

S-GKDS-TL 协议的体系结构和组密钥更新消息的广播机制基本上与 S-GKDS 相似，但在 S-GKDS-TL 协议中，组用户的初始化、组密钥的更新以及组用户的动态参与机制和 S-GKDS略有不同，关键体现在组密钥的生成机制和组用户的动态参与机制上。因为 S-GKDS-TL协议不但保留了 S-GKDS 中显式的组用户动态参与机制，还提供了一种灵活的隐式动态用户参与机制。

4.5.3.1 S-GKDS-TL 组用户的初始化

与 S-GKDS 协议相似，在 S-GKDS-TL 组用户的初始化阶段，组管理中心 GC 同样需要从 $F_q[x]$中随机地选取 m 个度为 t 的屏蔽多项式：

$$\{h_j(x) = h_{0,j} + h_{1,j}x + \cdots + h_{t,j}x^t\}_{j=1,2,\cdots,m} \in F_q[x] \tag{4.5.11}$$

另外，为保证组密钥的后向和前向隐私性，S-GKDS-TL 的 GC 需预先计算一条双向哈希链 DDHC：前向哈希链 $K^F=\{K_o^F, H(K_o^F), \cdots, H^{m-1}(K_o^F)\}$和后向哈希链 $K^B=\{H^{m-1}(K_o^F), H^{m-2}(K_o^F), \cdots, K_o^F\}$；另外，为了提供密钥更新消息的自愈机制，GC 还需预先计算一个密钥更新消息 RK 的单向哈希序列$\{\mathrm{RK}_i \mid i=1,2,\cdots,m\}$，并满足 $\mathrm{RK}_i=H(\mathrm{RK}_{i+1})$，$0\leqslant i\leqslant m-1$。

随后，GC 利用多项式$\{h_j(x)\}_{j=1,2,\cdots,m}$为每个在初始化阶段加入活动组的用户 U_i 产生 m 个秘密私钥 $S_i=\{h_1(i), h_2(i), \cdots, h_m(i)\}$，并通过 InitGroupKey 消息将这 m 个秘密私钥$\{h_1(i), h_2(i), \cdots, h_m(i)\}$以及会话 1 所对应的双向链 K^F 和 K^B 上的哈希值$\{H(K_o^F), H^{m-1}(K_o^B)\}$通过安全可靠的信道预先分发给用户 U_i。

$$\mathrm{GC} \rightarrow U_i : \{E_{\mathrm{MK}_i}(b \mid \mathrm{RK}_{b+2} \mid T_{\mathrm{refresh}} \mid H(K_o^F) \mid H^{m-1}(K_o^B) \mid h_1(i) \mid \cdots \mid h_m(i)) \mid$$
$$\mathrm{MAC}(b \mid \mathrm{RK}_{b+2} \mid T_{\mathrm{refresh}} \mid H(K_o^F) \mid H^{m-1}(K_o^B) \mid h_1(i) \mid \cdots \mid h_m(i))\},$$

其中，b 是密钥更新消息 RK 的缓冲区长度；T_{refresh}是组密钥更新周期，即组会话间隔时间；MAC(·)是产生消息验证码的散列函数(如 SHA1，MD5)；MK_i 是用户 U_i 与 GC 共享的主密钥，用于 InitGroupKey 消息的加密和验证。

U_i 一旦接收到 InitGroupKey 消息，首先验证该消息的真伪；若为真，则 U_i 利用 MK_i 解密该消息并获得相应的秘密私钥 $S_i=\{h_1(i), h_2(i), \cdots, h_m(i)\}$，双向哈希链 K^F 和 K^B 上的哈希值$\{H(K_o^F), H^{m-1}(K_o^B)\}$，以及密钥更新消息 RK_{b+2}。随后，用户利用单向函数 H 计算其他哈希序列值$\{\mathrm{RK}_j\}_{j=1,2,\cdots,b+1}$；复制 RK 消息序列到它对应的密钥更新消息缓冲区和密钥更新消息槽；计算组密钥$\{\mathrm{TEK}_j=f(H^j(K_o^F), H^{m-j}(K_o^B), \mathrm{RK}_j)\}_{j=1,2}$并复制到对应的组密钥槽中；最后指定 TEK_1 为当前活动的组密钥。在完成这些初始化处理后，组用户通过接收随后的密钥更新消息 RefreshKey 来同步地更新组密钥。其他详细的组用户初始化和组密钥的更新机制可参见图 4.4.4 和算法 4.4.1。

4.5.3.2 S-GKDS-TL 组密钥的恢复机制

S-GKDS-TL 组密钥更新消息的广播机制与 S-GKDS 基本一致。在用户初始化后，GC

将周期性地将密钥更新消息$\{RK_i | RK_{i-1}=H(RK_i), i=1,2,\cdots,m\}$按逆序依次分发给所有的活动用户，即 RK_1 释放给会话 1，RK_2 释放给会话 2，…，RK_m 释放给会话 m 等。因此，给定哈希链中的 RK_j，用户能够使用单向函数 H 来计算以前的密钥$\{RK_i | 1\leqslant i\leqslant j\}$，却不能计算出其他的密钥$\{RK_i | j+1\leqslant i\leqslant n\}$。

随后，对于 GC 的第 j 次组密钥更新，GC 将广播如下的组密钥更新消息 RefreshKey（或称为 B_j）给所有活动用户：

$$GC\rightarrow * : \{w_j(x) \mid \{R\} \mid MAC(\{R\} \mid w_j(x))\},$$

其中，多项式 $w_j(x)=g_j(x)\cdot RK_j+h_j(x)$，$h_j(x)$是屏蔽多项式；$RK_j$ 是当前的组密钥更新消息；$R=R_j\cup R_{j-1}\cup\cdots\cup R_1(|R|\leqslant t, R\cap G_j=\varnothing)$是在会话 j 前（包括 j）被撤销的所有用户集；多项式 $g_j(x)$按如下方式构造：

$$g_j(x)=\prod_{r_i\in R}(x-r_i) \tag{4.5.12}$$

一旦收到 RefreshKey 消息，活动的组用户将进行组密钥的更新或密钥的恢复（若在此之前存在丢失的密钥更新消息包）。S-GKDS-TL 组密钥的恢复机制与 S-GKDS 协议相似，算法 4.4.3 详细描述了组用户对消息组更新消息 RefreshKey 的处理过程。对某一特定的活动用户 $U_i\in G_i$ 而言，当收到 GC 广播的组密钥更新消息 B_j 时，它首先计算多项式 $w_j(x)$ 和 $g_j(x)$在点 i 处的值 $w_j(i)$和 $g_j(i)$。由于 $U_i\notin R$，故有 $g_j(i)\neq 0$，因此，用户 U_i 可以利用它在初始化阶段保留的秘密私钥 $h_j(i)$以及 $w_j(i)$和 $g_j(i)$进一步恢复出当前的组密钥更新消息 RK_j。

$$RK_j=(w_j(i)-h_j(i))/g_j(i) \tag{4.5.13}$$

随后，活动用户 $U_i\in G_i$ 可按下式计算当前的组通信密钥：

$$TEK_j=f(H^j(K_o^F), H^{m-j}(K_o^B), RK_j)。$$

另一方面，对于 $R=R_j\cup R_{j-1}\cup\cdots\cup R_1(|R|\leqslant t)$中被 GC 撤销的用户 U_r 而言，由于$\{g_j(r)=0 | \forall U_r\in R\}$，$w_j(r)=g_j(r)\cdot RK_j+h_j(r)=h_j(r)$，因此，即使他们同谋（$|R|\leqslant t$）也难以计算出当前的组更新消息 RK_j；进而也难以计算与之对应的组通信密钥 $TEK_j=f(H^j(K_o^F), H^{m-j}(K_o^B), RK_j)$。

组密钥 TEK 和密钥更新消息能够同步地进行更新。由于 RK 序列的单向性，接收者可以通过验证 $H^{e+1}(RK_j)\neq kb[b-e]$来确认 RK 是否属于相同的组密钥更新消息序列。

由于在组密钥的更新仅需要处理低开销的哈希运算，因此组密钥更新操作的计算开销并不大。同时，组密钥更新消息的隐性认证机制使得 GC 和用户之间无需消息重传，这也极大地减少了协议通信开销。

4.5.3.3 S-GKDS-TL 组密钥的自愈机制

单向哈希链为组密钥的更新提供了一种有效的自愈方法。正如上文所提到的，每一个有效用户都能够在会话 $j(1\leqslant j\leqslant m)$计算出 TEK：

$$TEK_j=f(H^j(K_o^F), H^{m-j}(K_o^B), RK_j),$$

其中，$H^j(K_o^F)$和 $H^{m-j}(K_o^B)$并不要求在密钥更新消息中传输，因为每一个活动用户都能够通过预先分发的哈希值$\{H^{j_1}(K_o^F), H^{m-j_2}(K_o^B)\}$独立计算出 $H^j(K_o^F)$和 $H^{m-j}(K_o^B)$；而 RK_j 则被封装在密钥更新消息 RefreshKey 中，由 GC 周期性地分发给所有活动用户。因此，只

要能够保证更新消息中 RK 的自愈性，就能保证组密钥的自愈更新。

为此，在 S-GKDS-TL 中必须提供一种机制，使得组密钥更新消息能够在广播信道上可靠地传输。与 S-GKDS 相似，S-GKDS-TL 组密钥的自愈机制同样依赖于哈希链的单向性。在初始阶段，GC 必须先选择一个随机数 RK_m 作为哈希链的秘密种子值，随后重复执行哈希函数 H，利用公式 $RK_i=H(RK_{i+1})$，$1\leqslant i\leqslant m-1$ 来计算剩下的哈希值，最终获得哈希链 $\{RK_i | i=1,2,\cdots,m\}$。

在随后的密钥更新阶段，所有的 $\{RK_i | 1\leqslant i\leqslant m-1\}$ 将由 GC 按逆序分发给所有的用户，即 RK_1 释放给会话 1，RK_2 释放给会话 2，…，RK_m 释放给会话 m 等。给定哈希链中的 RK_j，用户仅能够使用单向函数 H 来计算以前的密钥 $\{RK_i | 1\leqslant i\leqslant j\}$，却不能计算出其他的密钥 $\{RK_i | j+1\leqslant i\leqslant n\}$。

因此，尽管密钥更新消息很可能在传输过程中丢失，但组密钥更新信息的哈希链却能够提供一种有效的自愈机制。因为组用户可以通过单向哈希函数和最近接收到的 RK 来恢复先前丢失的密钥更新消息 RK，进而能够成功地通过计算获得当前的组密钥 TEK。考虑到 S-GKDS-TL 和 S-GKDS 协议的相似性，这种自愈算法同样能够有效地容忍较高的丢包率和错误率。

4.5.3.4 组用户的动态参与机制

组用户的动态参与机制要求协议在用户加入或离开活动组的情况下，能够有效地保证组会话密钥的前向隐私性和后向隐私性。

用户离开：S-GKDS-TL 协议在此同时提供了两种灵活的组用户撤销机制：显式用户撤销和隐式用户撤销。显式的组用户撤销机制与 B-GKDS 基本一致，是通过组密钥的更新消息来实现的。

（1）显式用户撤销：假设在第 j 次会话中，GC 需要撤销用户 U_r，则只需要 U_r 包括在广播消息 B_j 的 $\{R\}$ 集合中，即 $U_r\in R$。由于 $\{g_j(r)=0 | \forall U_r\in R\}$，用户 U_r 无法利用广播的密钥更新消息 B_j 及其先前保存的秘密 $h_j(r)$ 去计算当前的组密钥更新消息 RK_j，自然也就无法计算组密钥 TEK_j。

（2）隐式用户撤销：DDHC 的后向哈希链 K^B 保证了组密钥的前向隐私性。对活动周期为 $[j_1,j_2]$ 的组用户而言，它将在会话 j_2 后离开活动组。后向哈希链 K^B 的单向性，使得组用户能使用预分配的哈希值 $H^{m-j_2}(K_o^B)$ 来计算 K^B 中对应的在会话 j_2 以前（含会话 j_2）的哈希序列 $\{H^{m-j}(K_o^F) | 1\leqslant j\leqslant j_2\}$，其中 $H^{m-j}(K_o^B)=H^{j_2-j}(H^{m-j_2}(K_o^B))$；而它难以计算在会话 j_2 以后的哈希值序列 $\{H^j(K_o^B) | j_2<j\leqslant m\}$，进而，用户不可能计算会话 j 以后的组密钥 $TEK_j=f(H^j(K_o^F),H^{m-j}(K_o^B),RK_j)$，其中 $(j>j_2)$。我们称这是一种隐性的限时用户撤销机制，因为一旦 U_i 的活动周期过期 $(j>j_2)$，它将自动退出通信组而不需要 GC 的直接干涉。

用户加入：当用户 U_v 希望在会话 j 加入活动组，其相应的处理与组用户在 S-GKDS-TL 协议初启时的初始化过程相似，即：

（1）用户 U_v 首先需要从 GC 获得加入活动组的许可；如果成功，U_v 建立一个与 GC 共享的主密钥 MK_i。随后 GC 为 U_v 产生 $m-j+1$ 个秘密私钥 $\{h_j(v),h_{j+1}(v),\cdots,h_m(v)\}$，并通过 InitGroupKey 消息将秘密私钥 $\{h_j(v),h_{j+1}(v),\cdots,h_m(v)\}$ 和与会话 j 对应的哈希链

K^F 上的值 $H^j(K_o^F)$通过安全、可靠的信道分发给用户 U_v：

$$GC \to U_v : \{E_{MK_v}(b \mid RK_{b+2} \mid T_{refresh} \mid H^j(K_o^F) \mid h_j(v) \mid \cdots \mid h_m(v)) \mid$$

$$MAC(b \mid RK_{b+2} \mid T_{refresh} \mid H^j(K_o^F) \mid h_j(v) \mid \cdots \mid h_m(v))\},$$

其中，共享的主密钥 MK_v 用于 InitGroupKey 消息的加密和验证；$H^j(K_o^F)$是 U_v 在前向哈希链中与会话 j 对应的哈希值；b 是密钥更新消息 RK 的缓冲区长度；$T_{refresh}$是密钥更新周期。

(2) 用户 U_v 一旦接收到 InitGroupKey 消息，则按照算法 4.4.1 处理该消息，然后加入到活动组通信，接收随后的密钥更新消息 RefreshKey，并同步地更新组密钥 TEK，如算法 4.4.3 所示。

4.5.4 安全性分析

这里，我们对 S-GKDS-TL 协议进行与 S-GKDS 相似的安全性分析。在如下的推导过程中，我们用随机变量 $\boldsymbol{K}_j$ 表示对应的组密钥 TEK_i。

定理 4.5.1 组密钥管理协议 S-GKDS-TL 能够同时安全地主动撤销最多 t 个用户；并且，就信息论范畴而言，S-GKDS-TL 协议是无条件安全的。

证明：依据定义 4.5.1，假定 $\boldsymbol{U}=\{U_1,U_2,\cdots,U_n\}$是所有可能的组用户集；$R_j\subseteq\boldsymbol{U}$ 是第 j 次会话中由 GC 所撤销的组用户；$R=R_j\cup R_{j-1}\cup\cdots\cup R_1(|R|\leqslant t)$，是在会话 j 被撤销的所有组用户集；$J_j\subseteq\boldsymbol{U}$ 是第 j 次会话中新加入的组用户；$G_j=(G_{j-1}\cup J_j)\backslash R_j$ 是第 j 次会话中合法的组成员。

对比定义 4.5.1 和定义 4.4.1 可知，组密钥分发模型 $\boldsymbol{D}_{S\text{-}TL}(\boldsymbol{U},t,m)$的性质(1)～性质(5)完整地继承了 $\boldsymbol{D}_S(\boldsymbol{U},t,m)$的相关特性；并在此基础上，定义 4.5.1 的性质(6)引入了组密钥的时限访问特性。S-GKDS-TL 协议完全是 S-GKDS 协议的扩展，它具有 S-GKDS 协议的安全特性。因此，S-GKDS-TL 协议满足 $\boldsymbol{D}_{S\text{-}TL}(\boldsymbol{U},t,m)$模型所定义的性质(1)～性质(5)。如下只需证明 S-GKDS-TL 协议能够满足 $\boldsymbol{D}_{S\text{-}TL}(\boldsymbol{U},t,m)$所定义的性质(6)，即组用户的时限访问特性。

事实上，基于双向哈希链 DDHC 的限时用户撤销算法保证了活动用户的时限访问特性。假设活动周期为$[j_1,j_2]$的用户 $U_i\in G_r$ 在会话 j_1 加入会话，则在活动周期内，U_i 可以使用预先分配的种子值$\{H^{j_1}(K_o^F),H^{m-j_2}(K_o^B)\}$来计算在 DDHC 中$[j_1,j_2]$之间的哈希值 $H^j(K_o^F)$和 $H^{m-j}(K_o^B)$；然后，U_i 可以根据它接收的组密钥更新消息 B_j来计算多项式 $w_j(x)$和 $g_j(x)$在点 i 处的值 $w_j(i)$和 $g_j(i)$；并进一步利用它在初始化阶段保留的秘密私钥 $h_j(i)$以及 $w_j(i)$和 $g_j(i)$恢复出当前的组密钥更新消息 $RK_j=(w_j(i)-h_j(i))/g_j(i)$；最后，用户 $U_i\in G_i$ 可以计算出会话 j 对应的组密钥 $TEK_j=f(H^j(K_o^F),H^{m-j}(K_o^B),RK_j)$。

然而，当 $j<j_1$ 或 $j>j_2$ 时，用户 U_i 则不能有效地通过计算得到组密钥 TEK_j，因为双向哈希链 DDHC 的单向性使得用户 U_i 很难计算在它加入组会话 j_1 之前的 $H^j(K_o^F)(j<j_1)$，以及在它离开组会话 j_2 之后的 $H^{m-j}(K_o^B)(j>j_2)$，即组用户 U_i 被限制在$[j_1,j_2]$内访问 DDHC 双向链之间的哈希值。

因此，活动周期为$[r,s]$的用户 $U_i\in G_r$ 在会话 r 加入组通信后，它仅能获得$[r,s]$之间的组密钥 K_j，$r\leqslant j\leqslant s$；而无法获得$[1,r-1]$和$[s+1,m]$之间的组密钥，即有下式成立：

$H(\boldsymbol{K}_1,\cdots,\boldsymbol{K}_{r-1},\boldsymbol{K}_{s+1},\cdots,\boldsymbol{K}_m \mid \boldsymbol{B}_1,\cdots,\boldsymbol{B}_m,\{\boldsymbol{S}_i\}_{U_i\in G}) = H(\boldsymbol{K}_1,\cdots,\boldsymbol{K}_{r-1},\boldsymbol{K}_{s+1},\cdots,\boldsymbol{K}_m)$。□

定理 4.5.2 组密钥管理协议 S-GKDS-TL 能够保证组密钥的前向隐私性和后向隐私性。

在此,我们不再严格地证明该定理,仅作一般意义上的阐述。事实上,从定理 4.5.1 的证明过程可知,$\boldsymbol{D}_S(\boldsymbol{U},t,m)$模型的性质(6)已经隐含了组密钥的前向和后向隐私性。也即,双向哈希链 DDHC 的限时用户撤销算法有效地保证了 S-GKDS-TL 协议中组密钥的前向/后向隐私性。

4.5.5 性能分析

与 S-GKDS 相比,S-GKDS-TL 协议仅引入了时限的组用户撤销功能。然而在具体的实现机制上,S-GKDS-TL 协议则仅增加了一条后向哈希链 K^B 来保证时限用户撤销机制的实现。因此,GKDS-TL 协议和 S-GKDS-TL 协议具有相似的性能。在此,限于篇幅,我们不再作详细而类似的讨论。

表 4.5.1 分别对 S-GKDS-TL 协议、S-GKDS 协议、Stadden 的自愈分发协议以及 Liu-Ning 的自愈分发协议进行了性能对比分析。

表 4.5.1 性能对比

	通信开销(组播)	通信开销(单播)	存储开销
S-GKDS	$O(t\log q)$	$O(m\log q)$	$O(m\log q)$
S-GKDS-TL	$O(t\log q)$	$O(m\log q)$	$O(m\log q)$
Stadden[10]	$O((mt^2+mt)\log q)$	$O(m^2\log q)$	$O(m^2\log q)$
Liu-Ning[60]	$O((mt+m+t)\log q)$	$O(m\log q)$	$O(m\log q)$

存储开销:在初始化阶段,每个组用户 U_i 需要存储自己的身份标识 i 和掩码多项式 $\{h_j(x)\}_{j=1,2,\cdots,z}\in F_q[x]$在点 i 处的值$\{h_1(i),h_2(i),\cdots,h_z(i)\}$。因此,对所有用户 U_i 而言,其平均存储复杂度为 $O(m\log q)$。因此,就存储开销而言,S-GKDS 和 S-GKDS-TL 协议与 Liu-Ning 的自愈协议基本一样,都为 $O(m\log q)$;而 Stadden 的自愈协议则为 $O(m^2\log q)$,其存储优化的效果是显著的。

通信开销:广播消息 $\boldsymbol{B}_j$ 包括在第 j 次会话中由 GC 所撤销的组用户标识集 R 和一个 t 维多项式 $w_j(x)$;因此,S-GKDS 和 S-GKDS-TL 协议的通信开销都是 $O(t\log q)$,而 Stadden 的组密钥分发协议则为 $O((mt^2+mt)\log q)$,Liu-Ning 则为 $O((mt+m+t)\log q)$。显然,S-GKDS-TL 协议和 S-GKDS 协议使得 GC 和组用户之间的广播通信开销得到了显著优化,因为组密钥广播消息包的大小被减少到 $O(t\log q)$。特别地,由于 S-GKDS-TL 协议的通信开销独立于组通信的最大会话次数 m,只与撤销的最大用户数 t 相关,这使得当 m 较大时,协议的优化效果将更为显著。

4.5.6 时限用户撤销机制的改进

基于 DDHC 的时限用户撤销机制能够满足定义 4.3.6 所给出的前向/后向隐私性。然

而，协议还不能完全抵御组用户节点之间在某些情形下的同谋攻击。事实上，组密钥管理协议不仅要防止某个节点破解系统，还要防止某几个节点联合起来破解。即协议应具备良好的防同谋破解，以期杜绝同谋破解或尽量降低同谋破解的概率。

4.5.6.1　单向哈希二叉树

为了改善协议的计算效率(减少哈希运算次数)，我们可以将前述的限时组用户撤销机制中的双向哈希链 DDHC 替换为单向哈希二叉树(hash binary tree，HBT)。即利用哈希二叉树 HBT 来产生组密钥更新消息的 RK 序列，序列中每个组密钥更新消息分别与不同的时间段相对应。所有的组密钥更新消息 RK 与该哈希二叉树中的叶子节点相关联，由同一种子值经过哈希运算产生。

图 4.5.2 所示的单向哈希二叉树 HBT 所产生的哈希序列能够满足最大会话次数为 2^d 的组通信，其根种子值为 $S(0,0)$，树的深度为 $d=3$。算法 4.5.1 则描述了单向哈希二叉树的构造方法，值 $S(i,j)$ 中的 i 表示哈希种子在哈希二叉树中的深度，j 则表示种子在该层中的横向编号。

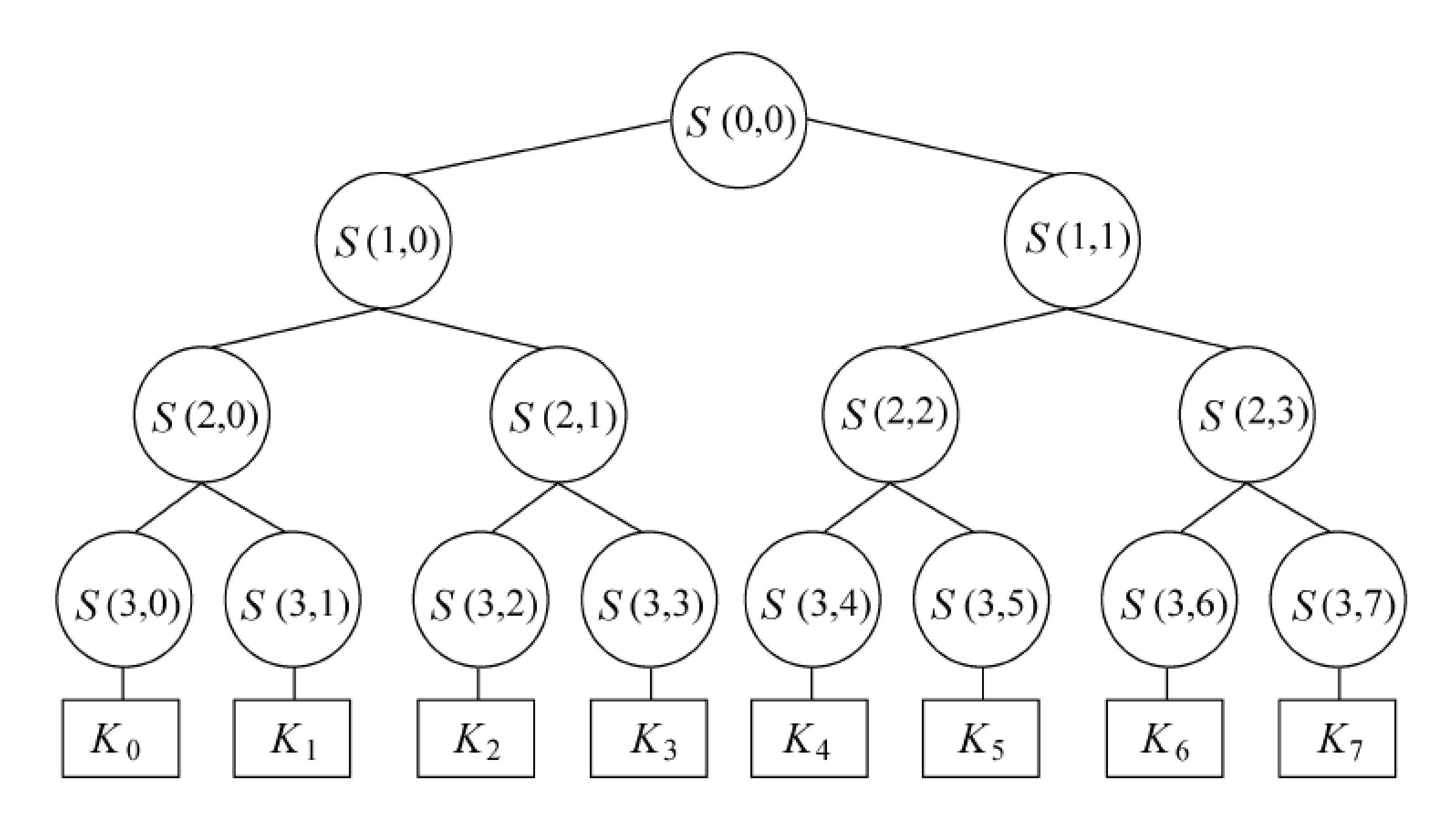


图 4.5.2　单向哈希二叉树($d=3$)

算法 4.5.1　哈希二叉树的生成算法

```
Function Generate_Hash_Binary_Tree(d){
  h=0;k=0;/*哈希种子值集合 Seed 的初始化*/
  产生哈希树的根种子 S(0,0);
  while (h<d){
    for (k=0;k≤2^h−1;k++){/*生成左右两个孩子对应的哈希种子值*/
      S(h+1,2k)=H(LeftShift(S(h,k)));/*生成 S(i,j)左孩子的种子值*/
      S(h+1,2k+1)=H(RightShift(S(h,k)));/*生成 S(i,j)右孩子的种子值*/
    }
    h=h+1;
  }
}
```

算法 4.5.1 首先要根据组通信的最大会话次数 m 来确定哈希二叉树的深度。为方便描述，我们假定组通信的最大会话次数为 $m=2^d$，与之对应的哈希二叉树深度则为 d。随后，GC 利用哈希树的根种子 $S(0,0)$ 依次生成左右两个孩子对应的种子：左移种子 $S(i,j)$ 一位，然后进行哈希运算得到左边孩子对应的种子 $S(i+1,2j)=H(\text{LeftShift}(S(i,j)))$；右移种子 $S(i,j)$ 一位，然后进行哈希运算，便可得到右边孩子对应的种子 $S(i+1,2j+1)=H(\text{RightShift}(S(i,j)))$。重复如上步骤，直到树的深度达到 d 即可生成哈希二叉树。最后，叶子节点 $S(d,0)$ 将对应于会话 1 的组密钥更新消息 RK_1，$S(d,1)$ 将对应于会话 2 的组密钥更新消息 RK_2，…，$S(d,2^d-1)$ 将对应于会话 $m=2^d$ 的组密钥更新消息 RK_m。

4.5.6.2 哈希二叉树种子值的预分发

当一个组用户 U_i 在会话 j_1 加入到一个活动组时，GC 将对该用户进行初始化，它根据用户 U_i 的活动周期 $[j_1,j_2]$ 分配哈希二叉树 HBT 上对应的哈希种子值给该用户。算法 4.5.2 描述了如何根据组用户 U_i 的活动周期 $[j_1,j_2]$ 来产生和哈希树对应的组密钥更新种子值。

算法 4.5.2 计算哈希二叉树的种子值

```
Function Generate_HBT_Seed_Value_for_User(j1,j2){
  l_child=j1-1,r_child=j2-1;
  l=l_child/2,r=r_child/2,h=d;
  Seed={S(h,l_child),S(h,r_child)};/*哈希种子值集合 Seed 的初始化*/
  while (l<r){
    if (S(h,l_child) ≠H(leftShift(S(h-1,l))))/*处理左子树*/
      Seed=Seed∪{S(h-1,l+1)};/*S(h,l_child)是 S(h-1,l)的右孩子节点*/
    else Seed=Seed\{S(h,l_child)};
    if (S(h,r_child) ≠H(RightShift(S(h-1,r))))/*处理右子树*/
      Seed=Seed∪{S(h,r-1)};/*S(h,r_child)是 S(h-1,r)的左孩子节点*/
    else Seed=Seed\S(h,r_child)};
    l_child=l, r=r_child;
    l=l/2, r=r/2;
    h=h-1;
  }
  return Seed;
}
```

直观上来看，对活动周期为 $[j_1,j_2]$ 的用户 U_i 而言，会话 j_1 对应的组密钥更新消息和 HBT 树的叶子节点 $S(d,j_1-1)$ 相关联；会话 j_2 所对应的组密钥更新消息和 HBT 树的叶子节点 $S(d,j_2-1)$ 相关联。算法 4.5.2 的主要策略是 GC 将哈希二叉树 HBT 树上的如下哈希值(有必要进一步消除冗余的哈希种子值)通过安全的信道预先分发给组用户 U_i：①叶子节点 $S(d,j_1-1)$ 和 $S(d,j_2-1)$ 所对应的哈希值；②从叶子节点 $S(d,j_1-1)$ 到根节点的路径上的所有节点的右兄弟节点(节点 $S(1,1)$ 除外)的哈希值；③从叶子节点 $S(d,j_2-1)$ 到根节点的路径上的所有节点的左兄弟节点(节点 $S(1,0)$ 除外)的哈希值。算法 4.5.2 中

的第 06～第 11 行包含了这种冗余处理机制，这样可以进一步减少节点存储的哈希种子值。

显然，HBT 树的单向性使得活动周期为$[j_1,j_2]$的组用户只能同时访问在 $j_1\leqslant j\leqslant j_2$ 范围内的组密钥更新值$\{RK_j|j_1\leqslant j\leqslant j_2\}$；而无法访问在范围$[1,j_1-1]$和$[j_2+1,m]$之间的组密钥更新值：$\{RK_j|1\leqslant j\leqslant j_1-1\}$和$\{RK_j|j_2+1\leqslant j\leqslant m\}$。

例如，对于图 4.5.2 中的哈希二叉树而言，若用户的活动周期是(2,5)，则 GC 将哈希二叉树 HBT 上的哈希种子值$\{S(3,1),S(2,1),S(3,4)\}$通过安全的通道分发给该用户。种子值$\{S(3,2),S(3,3)\}$没有必要直接分发给用户，因为用户可以利用种子值 $S(2,1)$，通过哈希运算获得种子值$\{S(3,2),S(3,3)\}$。因此，基于 HBT 的组用户撤销机制可以在存储和计算效率之间取得一种平衡。

与基于 HHDC 的 S-GKDS-TL 协议不同，在基于 HBT 树的 S-GKDS-TL 协议中，第 j 次会话的组通信密钥被定义为 j，$S(d,j)$和 RK_j 的函数：

$$TEK_j = f(S(d,j),RK_j) \tag{4.5.14}$$

其中，$1\leqslant j\leqslant m$；$S(d,j)$是和会话 j 对应 HBT 树叶子节点的种子值；RK_j 来源于 GC 的第 j 次组密钥更新消息，其更新处理机制与前述 S-GKDS-TL 相似。

4.5.6.3 基于 HBT 树的安全性分析

总体而言，基于 HBT 的组密钥分发协议具有比基于 DDHC 的组密钥分发协议更强的安全特性。

定理 4.5.3 基于 HBT 的 S-GKDS-TL 协议具有良好的抗同谋破解性。假定 $B\subseteq R_r\cup R_{r-1}\cup\cdots\cup R_1$ 为在会话 r 前撤销的组用户集，$F\subseteq J_s\cup J_{s+1}\cup\cdots\cup J_m$ 为在会话 s 后加入组通信的组用户集，则 $B\cup F$ 中的任何组用户子集的密谋均无法获得组通信密钥 $K_j(r\leqslant j\leqslant s)$。

证明：考虑任意用户子集 $B,C\subset\{U_1,U_2,\cdots,U_n\}$，其中 $B\subseteq R_r\cup R_{r-1}\cup\cdots\cup R_1$ 为在会话 r 前撤销的用户集，$F\subseteq J_s\cup J_{s+1}\cup\cdots\cup J_m$ 为在会话 s 后加入组通信的用户集。下面我们证明 $B\cup F$ 中的任何用户之间的密谋均无法获得组通信密钥 $K_j(r\leqslant j\leqslant s)$。

不妨设集合 B 中的任一用户 $U_i^B\in\{U_1^B,U_2^B,\cdots,U_{|B|}^B\}$的活动周期是$[L_i^B,H_i^B]$，其中 $1\leqslant L_i^B<H_i^B,H_i^B<r$。可得，$B$ 中所有用户的最大可能活动区间为$[H_{\min}^B,H_{\max}^B]$，其中 $H_{\min}^B$ 和 $H_{\max}^B$ 分别为

$$H_{\max}^B = \max\{H_1^B,H_2^B,\cdots,H_{|B|}^B\} \tag{4.5.15}$$

$$L_{\min}^B = \min\{L_1^B,L_2^B,\cdots,L_{|B|}^B\} \tag{4.5.16}$$

同样，对用户集 F 中的任一用户 $U_j^F\in\{U_1^F,U_2^F,\cdots,U_{|B|}^F\}$，不妨设其活动周期是$[L_i^F,H_i^F]$，其中 $s<L_i^F<H_i^F,H_i^F\leqslant m$。相应地，集合 F 中所有用户的最大可能活动区间是$[H_{\min}^F,H_{\max}^F]$，其中 $H_{\min}^F$ 和 $H_{\max}^F$ 分别为

$$H_{\max}^F = \max\{H_1^F,H_2^F,\cdots,H_{|F|}^F\} \tag{4.5.17}$$

$$L_{\min}^F = \min\{L_1^F,L_2^F,\cdots,L_{|B|}^F\} \tag{4.5.18}$$

显然，所有用户 $U_i^B\in B$ 和 $U_j^F\in F$ 的活动周期均分别落在区间$[H_{\min}^B,H_{\max}^B]$和$[H_{\min}^F,H_{\max}^F]$内。考虑到用户 $U_i^B\in B$ 是在会话 r 前被撤销的用户，而用户 $U_j^F\in F$ 是在会话 s 后加入组通信的用户，故活动区间$[H_{\min}^B,H_{\max}^B]$和$[H_{\min}^F,H_{\max}^F]$互不重叠：$[H_{\min}^B,H_{\max}^B]\cap[H_{\min}^F,H_{\max}^F]=\varnothing$，并且有如下不等式成立：

$$1 \leqslant L_{\min}^{B} < H_{\max}^{B} \leqslant r < s \leqslant L_{\min}^{F} < H_{\max}^{F} \leqslant m \tag{4.5.19}$$

因此，若 Seed_B 和 Seed_F 分别表示集合 B 和 F 中所有用户被预先分发的哈希种子值集，则有 $\text{Seed}_B \cap \text{Seed}_F = \varnothing$。即所有用户 $U_i^B \in B$ 在 HBT 树上对应叶子节点的哈希种子值所组成的种子集合 Seed_B，与所有用户 $U_j^F \in F$ 在 HBT 树上对应叶子节点的哈希种子值所组成的种子集合 Seed_F 的交集为空。另外，由于 $H_{\max}^{B} \leqslant r < s \leqslant L_{\min}^{F}$，故有 $\{S(d,r), S(d,r+1), \cdots, S(d,s)\} \not\subset \text{Seed}_B \cup \text{Seed}_F$。

因此，根据哈希二叉树的构造算法 4.5.1 和算法 4.5.2，以及哈希树的单向性，我们可以得出如下结论：$B \cup F$ 中的任何用户之间的密谋均无法获得区间 $[r,s]$ 之内的哈希种子值 $\{S(d,j) \mid r \leqslant j \leqslant s\}$，进而无法获得该区间内的组密钥 $\{K_j \mid r \leqslant j \leqslant s\}$，即有：

$$H(\boldsymbol{K}_r, \boldsymbol{K}_{r+1}, \cdots, \boldsymbol{K}_{s-1} \mid \boldsymbol{B}_1, \cdots, \boldsymbol{B}_m, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{B}}, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{R}}) = H(\boldsymbol{K}_r, \boldsymbol{K}_{r+1}, \cdots, \boldsymbol{K}_{s-1}) \tag{4.5.20}$$

□

在 $\boldsymbol{D}_{\text{S-TL}}(\boldsymbol{U}, t, m)$ 模型中引入如上性质，则我们可以定义一个能够抵御组用户同谋破解的组密钥分发协议模型 $\boldsymbol{D}_{\text{S-TL}}^{*}(\boldsymbol{U}, t, m)$。

定义 4.5.2 假定 $\boldsymbol{U} = \{U_1, U_2, \cdots, U_n\}$ 是所有可能的组通信用户的集合，m 是组通信系统的最大会话次数，t 是 GC 所能主动撤销的最大用户数，则 $\boldsymbol{D}_{\text{S-TL}}^{*}(\boldsymbol{U}, t, m)$ 是一个改善的自愈组密钥分发模型，若如下条件能够满足：

(1) 对组用户 $U_i \in G_i$ 而言，组密钥 K_j 完全可由 B_j 及其私钥 S_i 决定，即

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j, \boldsymbol{S}_i) = 0 \tag{4.5.21}$$

(2) 对任何用户子集 $B \subseteq \boldsymbol{U}$，$|B| \leqslant t$，集合 B 中的用户不可能获得用户 $U_k \notin B$ 的用户私钥 S_k，即

$$H(\boldsymbol{K}_j, \boldsymbol{S}_k \mid \boldsymbol{B}_j, \boldsymbol{B}_{j-1}, \cdots, \boldsymbol{B}_1, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{B}}) = H(\boldsymbol{K}_j, \boldsymbol{S}_k) \tag{4.5.22}$$

(3) 不可能单独地从 GC 广播的密钥更新信息或组用户私钥获得组密钥，即

$$H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m \mid \boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_m) = H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m) \tag{4.5.23}$$

$$H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m \mid \boldsymbol{S}_1, \boldsymbol{S}_2, \cdots, \boldsymbol{S}_n) = H(\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_m) \tag{4.5.24}$$

(4) $\boldsymbol{D}_{\text{S-TL}}^{*}(\boldsymbol{U}, t, m)$ 能够同时安全撤销最多 t 个用户的能力：设每次会话 j 被撤销的组用户集是 $R = R_j \cup R_{j-1} \cup \cdots \cup R_1$ $(|R| \leqslant t)$，则 R 中的组用户不可能利用 GC 广播的组密钥更新信息 B_j 去恢复出当前的组通信密钥 K_j，即

$$H(\boldsymbol{K}_j \mid \boldsymbol{B}_j, \boldsymbol{B}_{j-1}, \cdots, \boldsymbol{B}_1, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{R}}) = H(\boldsymbol{K}_j) \tag{4.5.25}$$

(5) 对用户 $U_i \in G_r$，若其在会话 r 收到密钥更新消息 $\{B_r \mid 1 \leqslant r < m\}$，但在会话 s 前一直没有被撤销 $(r < s \leqslant m)$，则该用户能够利用会话 s 收到的密钥更新消息 $\{B_l \mid r \leqslant l \leqslant s\}$ 恢复所有的组通信密钥 $\{K_l \mid r \leqslant l \leqslant s\}$，即

$$H(\boldsymbol{K}_r, \boldsymbol{K}_{r+1}, \cdots, \boldsymbol{K}_s \mid \boldsymbol{B}_r, \boldsymbol{B}_s, \boldsymbol{S}_i) = 0 \tag{4.5.26}$$

(6) (时限访问特性)对活动周期为 $[r,s]$ 的用户 $U_i \in G_r$ 而言，其在会话 r 加入组通信，则 U_i 仅能获得 $[r,s]$ 之间的组密钥 $K_j (r \leqslant j \leqslant s)$；而无法获得 $[1, r-1]$ 和 $[s+1, m]$ 之间的组密钥，即

$$\begin{aligned} & H(\boldsymbol{K}_1, \cdots, \boldsymbol{K}_{r-1}, \boldsymbol{K}_{s+1}, \cdots, K_m \mid \boldsymbol{B}_1, \cdots, \boldsymbol{B}_m, \{\boldsymbol{S}_i\}_{U_i \in \boldsymbol{G}}) \\ & = H(\boldsymbol{K}_1, \cdots, \boldsymbol{K}_{r-1}, \boldsymbol{K}_{s+1}, \cdots, \boldsymbol{K}_m) \end{aligned} \tag{4.5.27}$$

(7)(抗同谋破解性)假定 $B\subseteq R_r\cup R_{r-1}\cup\cdots\cup R_1$ 为在会话 r 前撤销的用户子集,$F\subseteq J_s\cup J_{s+1}\cup\cdots\cup J_m$ 为在会话 s 后加入组通信的用户子集。若 $|B\cup F|\leqslant t$,则 $B\cup F$ 中的任何用户子集的密谋均无法获得组通信密钥 $\{K_j|r\leqslant j\leqslant s\}$,即

$$H(\boldsymbol{K}_r,\boldsymbol{K}_{r+1},\cdots,\boldsymbol{K}_{s-1}|\boldsymbol{B}_1,\cdots,\boldsymbol{B}_m,\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{B}},\{\boldsymbol{S}_i\}_{U_i\in\boldsymbol{R}})=H(\boldsymbol{K}_r,\boldsymbol{K}_{r+1},\cdots,\boldsymbol{K}_{s-1}) \quad (4.5.28)$$

定义 4.5.2 的性质(1)～性质(6)完整地继承了定义 4.5.1 的性质;而定义 4.5.2 中的性质(7)则描述了 $\boldsymbol{D}^*_{\text{S-TL}}(\boldsymbol{U},t,m)$ 是组密钥分发模型所应满足的重要安全属性——抗同谋破解性。

值得指出的是,$\boldsymbol{D}^*_{\text{S-TL}}(\boldsymbol{U},t,m)$ 模型具有比 Stadden 和 Liu-Ning 的自愈模型更强的安全性。事实上,我们消除了 Stadden 和 Liu-Ning 模型的附加约束条件:$|B\cup F|\leqslant t$。这是一个重要的改进。因此,基于 HBT 树的 S-GKDS-TL 协议在安全性方面是优于 Stadden 的自愈协议以及 Liu-Ning 的自愈协议的,因为基于 HBT 树的 S-GKDS-TL 具有更强的抗同谋攻击特性。

定理 4.5.4 基于 HBT 的组密钥管理协议 S-GKDS-TL 在信息论范畴内是无条件安全的。

证明:比较定义 4.5.2 和定义 4.5.1 可知,组密钥分发模型 $\boldsymbol{D}^*_{\text{S-TL}}(\boldsymbol{U},t,m)$ 的性质(1)～性质(6)完整地继承了 $\boldsymbol{D}_{\text{S-TL}}(\boldsymbol{U},t,m)$ 的相关特性;因此,模型 $\boldsymbol{D}^*_{\text{S-TL}}(\boldsymbol{U},t,m)$ 的性质(1)～性质(6)的证明可参见定理 4.5.1;至于性质(7)的证明,即证明基于 HBT 的 S-GKDS-TL 协议能够满足 $\boldsymbol{D}^*_{\text{S-TL}}(\boldsymbol{U},t,m)$ 所定义的抗同谋破解特性,可参见定理 4.5.3 的证明,这里从略。

4.5.6.4 HBT 和 DDHC 方法的性能比较

存储开销:总体而言,将限时组用户撤销机制中的双向哈希链 DDHC 替换为单向哈希二叉树 HBT,采用的策略是以存储换取计算效率。因为在 DDHC 方法中,组用户仅需要存储两个种子值:前向哈希链的种子值 K_0^F 和后向哈希链的种子值 K_0^B。

为方便分析,我们假定组通信的最大会话次数是 $m=2^d$。考虑在基于哈希树的组用户撤销方法中,对活动周期为 (j_1,j_2) 的用户 U_i 而言,组用户节点需要存储的哈希种子值的数目是:

(1) 最好情况:组用户节点仅需要存储 1 个哈希种子值,此时会话 j_1 在 HBT 树中所关联的叶子节点 $S(d,j_1-1)$ 和会话 j_2 在 HBT 树中所关联的叶子节点 $S(d,j_2-1)$ 具有共同的祖先节点;而在以该祖先节点为根节点的子树中,节点 $S(d,j_1-1)$ 和 $S(d,j_2-1)$ 分别是该子树的最左和最右的叶子节点。

(2) 最坏情况:当 $j_1=2,j_2=m-1$ 时,组用户需要存储 $2\cdot\lceil\log m\rceil-2$ 个哈希种子值。

(3) 在一般情况下,组用户需要存储的哈希种子值数目则介于 1 和 $2\cdot\lceil\log m\rceil-2$ 之间。

基于哈希二叉树的限时用户撤销机制还有另外一个显著的优点。组用户可以申请在多个互不重叠的时间段访问组密钥,例如 $[j_1,j_2]$ 和 $[j_3,j_4]$,这时,GC 仅需要根据算法 4.5.2 把相应的哈希种子值预先分发给该用户即可;而基于 DDHC 的方法则存在一定的局限性,它仅提供了用户访问一个特定时间段的机制。这种特性极大地增强了 GC 对组用户访问控制的灵活性。

计算开销:总体而言,HBT 方法比 DDHC 方法应该有更高的计算效率。在 HBT 方法中,用户节点获得组密钥更新消息所需要的最大哈希计算次数是 $\lceil\log m\rceil$,因为哈希树的最大深度是 $\lceil\log m\rceil$;最小计算次数是 1。

而 DDHC 方法的最大哈希计算次数是 $2(m-1)$ 次,最小计算次数也是 2。平均计算次

数可以分析如下：

不失一般性，我们考察活动周期为(j_1, j_2)的用户U_i。根据双向哈希链 DDHC 的构造可知，U_i能够使用预先分发的哈希种子值$\{H^{j_1}(K_o^F), H^{m-j_2}(K_o^B)\}$来计算$\{H^j(K_o^F) \mid j_1 \leqslant j \leqslant m\}$和$\{H^{m-j}(K_o^B) \mid 1 \leqslant j \leqslant j_2\}$。显然，用户在计算$[j_1, j_2]$之间的双向哈希值所用的哈希运算次数为$2(j_2 - j_1 + 1)$次。

若组用户的活动周期在$[1, m]$范围内均匀分布，则 DDHC 方法中哈希运算次数的期望值为

$$E(C_{\text{hash}}) = \frac{1}{m}(0 + 2 + 4 + 6 + \cdots + 2(m-1)) = m - 1 \qquad (4.5.29)$$

因此，若以哈希计算次数作为协议的计算复杂度的衡量标准，则 DDHC 方法的平均计算复杂度是$O(m)$。

4.6 协议的具体应用

在前面 3 节，我们的研究更多地限于协议本身的安全性和性能方面，没有讨论 S-GKDS 和 S-GKDS-TL 协议的具体应用。

无线通信技术的飞速发展，使无线网络成为当前研究的一个热点问题。无线网络具有与生俱来的广播通信特性，所以安全组通信的研究可以应用在无线网络环境。但无线网络环境下的安全组通信密钥管理的研究要考虑到无线网络较有线网络所具有的不同特点。无线网络相对于有线网络，其主要不同点是：网络带宽、网络传输介质的可靠性。因此，无线环境下的安全组密钥管理的研究应注意考虑提高可靠性和降低网络带宽开销。此外，移动网络组用户节点的移动特性(mobility)增加了组密钥管理机制研究的难度。具有良好适应性的移动组通信中的密钥管理方式应该能在不干扰网络无缝连接特性的情况下，可以在网络间移动，同时不会离开安全通信组。当用户从一个网络移动到另一个网络时，网络的可信任性和移动用户的切分(handoff)服务是安全组通信要考虑的关键问题。

下面结合具体网络应用环境的特点，如无线网络、移动网络(NEMO 网)和无线传感器网络，着重探讨 S-GKDS 协议和 S-GKDS-TL 协议在这些具体网络环境中的可能应用。

4.6.1 无线传感器网络

无线传感器网络(wireless sensor networks，WSN)[71~73]是由一组传感器节点通过无线介质连接构成的无线网络，它采用自组织方式配置大量微型的智能传感节点，通过节点的协同工作来采集和处理网络覆盖区域中的目标信息。无线传感器网络在环境与军事监控，地震与气候预测，地下、深水以及外层空间探索等许多方面都具有广泛的应用前景。可以说无线传感器网络是信息感知和采集的一场革命，是 21 世纪最重要的技术之一。

无线传感器网络体系结构由 3 个主要部分组成：传感节点、聚类首领(group head，GH)节点和终端(sink)节点。传感节点散布在观察区域内采集与观察对象相关的数据，并将协同处理后的数据传送到 sink 节点。而后 sink 节点可以通过 Internet 或通信卫星实现

传感器网络与任务管理节点通信。如果网络规模太大,可以采用聚类分层的管理模式。

相对于有线网络,无线传感器网络的这种开放式体系结构,使得它更易于受到各种不安全因素的威胁[74~76]。我们可以知道无线传感器网络的组通信协议设计应该考虑到如下的性能标准:

(1) 能源有效性/生命周期:能源有效性是无线传感器网络设计中要考虑的重要因素。尽可能降低节点的能源消耗,从而延长网络生命周期,是无线传感器网络中有关协议设计的重要目标。

(2) 可靠性/容错性:传感节点容易因为能源耗尽或环境干扰而失效。部分传感节点的失效不应影响整个网络的任务。

(3) 可扩展性:在一些应用中可能需要成百上千个传感节点,组密钥分发协议的设计应能满足大量节点协作。

(4) 时延性:传感器网络的延迟时间是指观察者发出请求到收到应答信息所需时间[77]。因此,协议的设计不但应该尽可能地减少时延,而且要能容忍传感节点之间的时钟在一定程度上的扭曲。

S-GKDS 协议和 S-GKDS-TL 协议能为大规模的传感器网络提供一种安全的组通信方式。为方便管理,可以将整个网络分成多个簇(cluster),每个簇包含一个聚类首领节点。GH 在此充当组通信控制中心 GC,而各个簇内的传感器节点则充当组用户节点。根据前述协议的有关特性可知,S-GKDS 和 S-GKDS-TL 协议能够完成如下基本安全目标:

(1) 机密性(confidentiality):防止通信数据被窃听,确保攻击者不能获得任何关于明文的信息。

(2) 数据完整性(integrity):防止通信数据被篡改。

(3) 访问控制(confidentiality):确保只有合法的用户才能访问网络。

(4) 组密钥的更新和撤销(revocability):保护网络以避免遭受入侵节点的危害。

其中,安全目标(4)的实现还依赖于入侵检测系统的合作。该主题超出了本节的研究范围,在此,我们不再详细讨论。唯一需要知道的是,设置在传感器网络中的入侵检测系统一旦检测到某个节点已经被非法入侵,则传感器网络的任务管理节点必须通过某种安全机制通知该节点所在簇的聚类首领 GH 节点,并通过 GH 的组密钥广播更新消息撤销该节点,最终完成组密钥的更新。

特别需要指出的是,相对于 S-GKDS 协议而言,S-GKDS-TL 协议更适于在无线传感器网络中的应用。考虑到 WSN 的节点具有明确的生命周期,对于未受侵害的传感器节点,我们可以采用时限的组用户撤销机制来管理。这种隐式的用户撤销方式使得用户在退出会话时不需要组管理中心 GC(即聚类首领 GH)的直接介入,而且对撤销用户的总数没有限制。前者能够显著减少节点和 GH 的计算和通信负载;后者则有效地解决了传感器网络中组通信协议的可扩展性问题。特别对一些可能需要成百上千个传感节点的大规模应用,该特性尤显重要。另一方面,S-GKDS-TL 协议的显式用户撤销机制能够有效地避免无线传感器网络被已遭受入侵的节点所危害。一旦发现某个组用户已受到入侵,该用户节点所在簇的聚类首领 GH 即可通过组密钥广播更新消息来撤销该用户,以确保只有合法的组节点才能访问网络。

当需要增加某簇中的传感节点数量时,S-GKDS 和 S-GKDS-TL 协议都能提供一种简

洁的组节点参与机制。而且，无论是显式用户撤销，还是隐式用户撤销机制，都是轻量级的运算，其仅需要用户节点做哈希运算。

因此，S-GKDS-TL 协议优良的动态性能，能为传感器网络中用户节点参与或退出会话的组通信提供良好的自适应性。时限用户撤销机制的引入使得 S-GKDS-TL 协议提供了两种高效的组用户撤销管理机制：其一是通过组密钥广播更新消息实现的显式用户撤销；其二是通过时限用户撤销机制实现的隐式用户撤销。这两种用户撤销机制为 S-GKDS-TL 协议提供了一种灵活而高效的组用户动态管理机制，使得该协议能够很好地适应传感器网络中节点拓扑结构频繁变化的组通信应用。

4.6.2 NEMO 组通信

相对于传统的无线移动通信而言，下一代无线系统应该提供给用户更高的宽带服务，并且透明地将技术集成到系统环境中，从而实现位置无关性。这样就需要整合异构网络和协议。无线个人网络(wireless personal networks，WPN)是这种异构体系结构中不可或缺的一部分。下一代 WPN 需要全球范围内的无缝连接，用户可以在任何时间、任何地点使用最适合的接口接入网络。作为 WPN 网络中的重要组成部分，IETF 的 NEMO(network mobility)工作组认为移动网是一个具有 Internet 接入点的独立单元，也可以认为它是一个叶子网络[78~82]。不过，无论是使用多个移动路由器还是使用具有多个接口的单个移动路由器，都有可能是多地址的。NEMO 不仅要求提供和主干网络连接的功能，而且还需要具有用户位置注册和用户位置发现的功能。因此，未来 WPN 具备的一个重要特征是用户的多地址和在多域环境中的移动性。移动用户在不同作用域之间移动时能够保持连接而且连接路径是最优的，这将产生额外的需求，如无线资源管理和最优连接线路切换。

传统的移动 IP 所提供的漫游机制，仅局限于主机(host)漫游的无线移动网络。当主机扩展成网络时，即为具有移动网络 NEMO 特性的网络。例如，在公交设施上设置一台移动路由器(mobile router)，此路由器通过上行接口对外连接 Internet 网；对内通过无线局域网向移动用户提供接入服务。当公共交通设施移动漫游时，其内的所有移动用户将被视为一个移动子网。此时，移动用户设备并不需要单独执行漫游和无线切分(handoff)服务，取而代之的是通过移动路由器执行漫游与切分服务来保持网络整体的畅通。IETF 工作小组所定义的 NEMO 体系结构如图 4.6.1 所示。

NEMO 网络是由一个或多个 IP 子网所构成，以移动网络(mobile network)当作一个漫游或者切分(hand off)服务单位，并通过移动路由器连接至 Internet 网络。在图 4.6.2 中，移动网络通过移动路由器连接至家乡网络(home link)，并在移动网内部连接移动子网节点(mobile network node，简称 MNN)。当移动子网节点 MNN 漫游到另一个移动网时，它通过移动路由器接入他乡网络(foreign link)，最终通过 Internet 接入路由器来完成漫游和切分服务的平滑迁移。

移动子网节点 MNN 可分为 3 类：本地固定节点(local fixed node，LFN)、本地移动节点(local mobile node，LMN)及访问移动节点(visiting mobile node，VMN)。

(1) 本地固定节点(LFN)：为固定的主机或路由器节点，在此移动网络环境中，不会改变与移动路由器的连接方式。

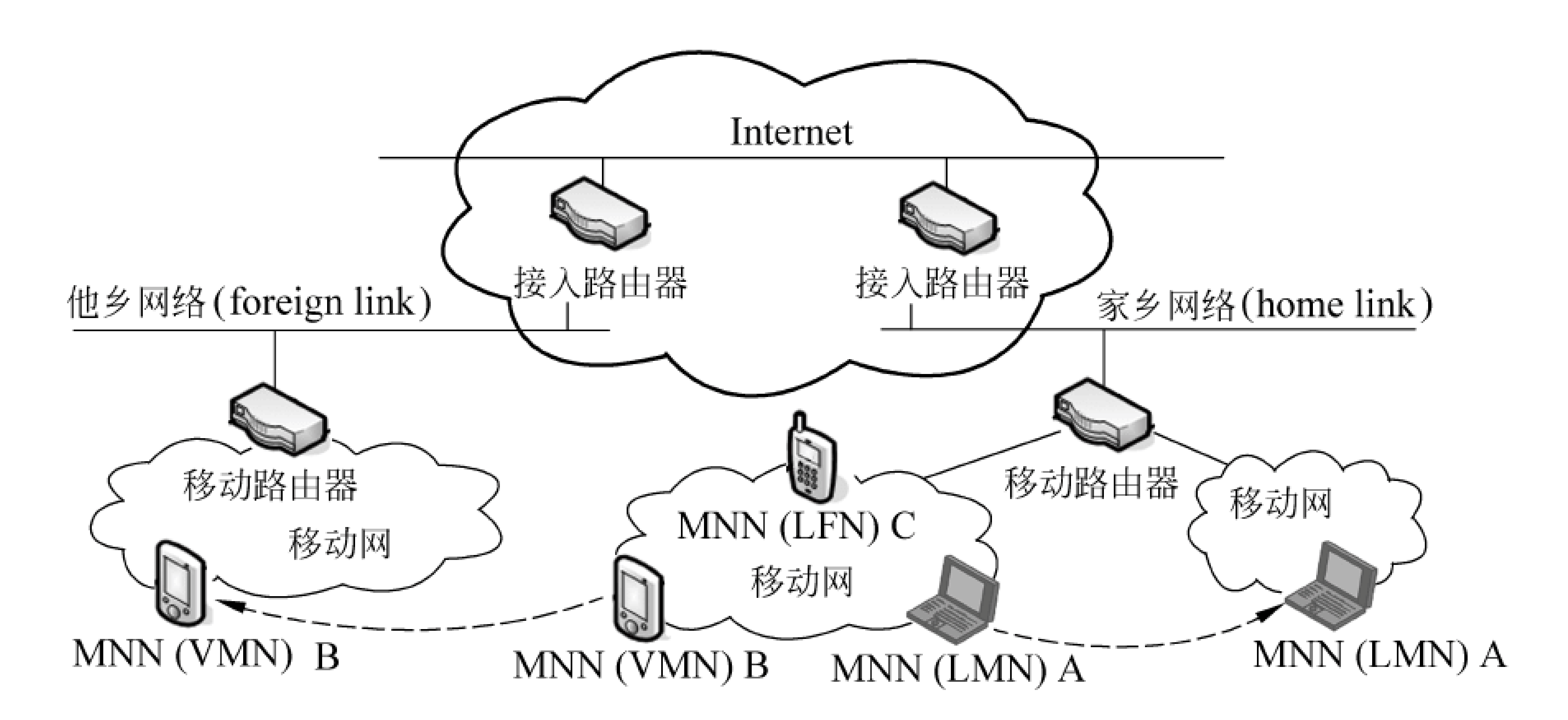


图 4.6.1 IETF NEMO 工作组定义的体系结构

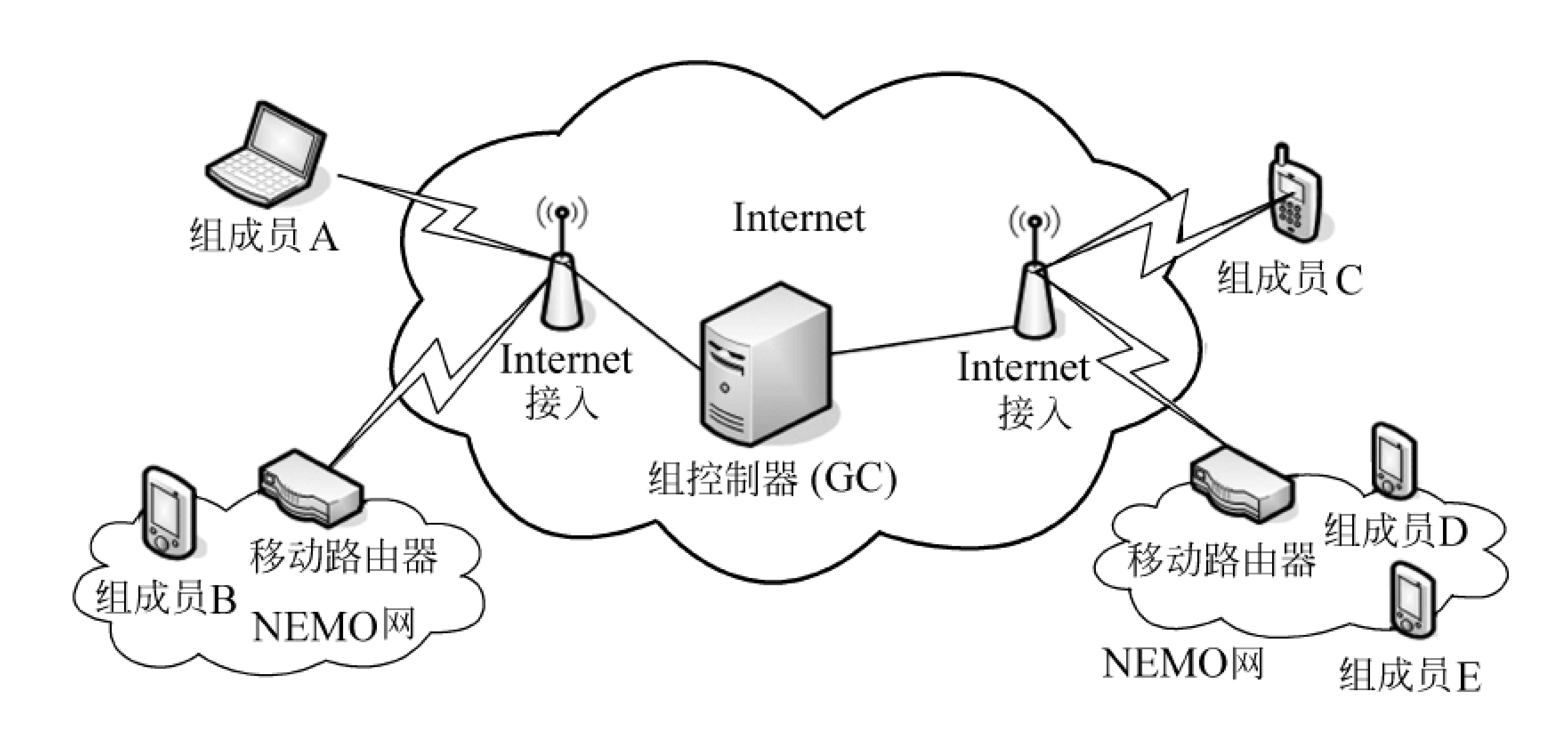


图 4.6.2 NEMO 环境下的组通信模型

（2）本地移动节点（LMN）：为移动的主机或路由器节点，在此移动网络环境中，节点可以自由地移动或漫游至其他子网，但不会跨越出此移动路由器所连接的子网络范围之外。

（3）访问移动节点（VMN）：为移动的主机或路由器节点，它可以自由地移动或漫游至其他子网，亦可移动或漫游跨越至其他移动路由器所在的移动网络中。

图 4.6.2 还解释了本地移动节点（LMN）和访问移动节点（VMN）的漫游机制：访问移动节点 VMN B 从家乡网络漫游到他乡网络，以及本地移动节点 LNN A 在同一移动网络中从其中一个子网漫游到另一个子网。

随着无线网络速度的提高，现有的技术包含 IEEE 802.11b 和 802.11g 等无线局域网（WLAN），最大速度皆可达到 11Mbps 以上，可满足一般的媒体视频带宽要求。因此，将无线 WLAN 协议 IEEE 802.11b/g 和 IPv6 相结合，进行 NEMO 环境下安全组通信的研究，提供以 IPv6 网络为基础的无线宽带网络环境下的普适计算服务，具有一定的现实意义。

结合组密钥分发协议和 NEMO 的通信机制，我们考察如下应用，如图 4.6.2 所示，NEMO 组通信会话涉及到 GC 和组成员。在此，GC 是一个移动的组密钥管理中心，全权负责组通信密钥的创建、分发和组成员关系发生变化时的密钥更新。而组成员则包含两种类型的用户：①孤立的组成员，例如，如图 4.6.2 中的节点 A 和 C；②处于 NEMO 网络中的分支节点，如图中的节点 B，D 和 E。所有成员节点以有线或无线的方式接入

Internet 网。

在组通信过程中，GC 和节点之间存在着多种控制信息（如图 4.4.2 所示）；用户节点的处理流程则按图 4.4.2 进行。即在用户节点上布置有消息验证和检查模块、组密钥更新消息 *RK* 的自愈模块、组密钥 TEK 切换模块以及流加密/解密和完整性检查模块。

4.6.3 进一步的研究工作

S-GKDS 和 S-GKDS-TL 协议是针对不可靠、易受攻击、开放的组通信环境而提出的组密钥分发协议。有关性能和安全性分析表明，S-GKDS 和 S-GKDS-TL 协议能够为无线传感器网络和 NEMO 这种异构网络环境下的组通信提供一种简洁而有效的组通信机制，其理由如下：

（1）S-GKDS 和 S-GKDS-TL 协议的组密钥广播更新机制使得在协议的交互过程中不再需要对丢失的消息进行重传或对接收的消息状态进行确认（ACKs）；这样能够有效地减少 GC 和组用户节点之间的通信开销。

（2）S-GKDS 和 S-GKDS-TL 协议能够有效地发现和恢复丢失的组密钥更新消息；这种组密钥的自愈机制使得协议能够容忍较高的信道丢包率和在一定程度上抵御 DoS 攻击。因此，S-GKDS 和 S-GKDS-TL 协议能够很好地适应无线传感器网络和 NEMO 移动网络这种开放、易受攻击、不可靠的无线通信环境。

（3）组更新消息中隐含的包验证机制有效避免了消息认证码（MAC）的构造和验证，这样能够有效地减少额外的计算和通信开销。

（4）S-GKDS 和 S-GKDS-TL 协议的组密钥的平滑更新机制使得在组密钥的切换过程中无需干扰正在进行的数据传输。

（5）S-GKDS 和 S-GKDS-TL 协议具有良好的可扩展性，能够适应较大规模的用户频繁地参与或退出会话的组通信环境；组密钥更新消息包的大小不受会话过程中被撤销的组用户数目的影响。

（6）S-GKDS 和 S-GKDS-TL 协议并不要求组通信中各组件（用户节点和 GC）之间的时钟完全精确同步，它能在一定程度上容忍时钟的不同步。这使得协议对 NEMO 这种时延较大的通信环境具有良好的自适应性；同样地，协议对于无线传感器网络这种分布式、缺乏网络时间同步机制的自治系统而言，也具有非常好的适应性。

（7）S-GKDS 和 S-GKDS-TL 协议在信息论范畴内是无条件安全的，它能够有效地保证组密钥的前向/后向隐私性；S-GKDS-TL 协议能够更有效地保证组密钥的抗同谋破解和组机密性。

值得一提的是，与 S-GKDS 协议相比，S-GKDS-TL 协议能够更好地适应无线传感器网络和 NEMO 移动网络环境下的组通信。因为，从 S-GKDS-TL 协议的构造机制来看，该协议是对 S-GKDS 协议的扩展，它完整地继承了 S-GKDS 协议的所有基本特性，如自愈性、安全性、自适应性和高性能。并且，它在 S-GKDS 协议的基础上，引入了一种基于时限的组用户撤销机制。这种轻量级的时限用户撤销机制并没有给 S-GKDS-TL 协议本身带来显著的计算和通信负载。因此，S-GKDS-TL 协议具有更优的可扩展性和动态性能，能为组用户频

繁参与或退出会话的组通信提供更好的自适应性。因为时限用户撤销机制使得组用户能以一种隐式方式退出组会话，这种机制具有如下特点：①用户的退出并不需要组管理中心的直接介入；②协议对撤销用户的总数没有限制。

S-GKDS-TL 协议提供了两种高效的组用户管理机制：其一是通过组密钥广播更新消息实现的显式用户撤销；其二是通过时限用户撤销机制实现的隐式用户撤销。这两种用户撤销机制的存在，使得协议能够提供一种灵活而高效的组用户动态管理机制，从而更好地适应传感器网络和 NEMO 环境下用户拓扑结构频繁变化的组通信应用。

我们针对组密钥分发协议的鲁棒性、可扩展性和动态稳定性等特性进行了一些有益的探索，并给出了一些创新性结论。但考虑到组通信应用环境的实际复杂性，还存在如下一些问题，需要进一步地进行深入研究。

首先，提出的 B-GKDS，S-GKDS 和 S-GKDS-TL 协议存在一个共同的局限性，即组会话的最大次数是 m；这种约束的存在一定程度地制约了协议的应用范围。对某些实际应用而言，我们需要考虑无会话次数限制的组密钥分发协议。因此，如何改进这些协议并消除这种约束，是我们目前正在进行的研究工作重点。值得一提的是，解决该问题的相关协议雏形正在形成，并处于逐步完善的过程中。

其次，关于安全组密钥管理的研究主要集中在组密钥分发机制本身的安全性、动态性和可扩展性方面的研究，存在着与实际应用结合得不够等问题。因此，我们有必要在将来的工作中结合具体应用环境的特点，如无线网络和 NEMO 移动网络，对 S-GKDS 和 S-GKDS-TL 协议进行逐步完善和求精，并对协议的实际性能、安全性进行相关测试分析，以期进一步突出研究成果的实用价值。

此外，组通信具有广泛的应用，把组密钥管理方案与其他组通信的应用特性相结合，寻求最佳的管理方案以保证数据传输所需的各种性能和安全特性，仍具有重要的研究意义和实用价值。因此，我们认为关于该主题的研究还可以在如下几个方向加以深入展开：

(1) 在安全的组密钥管理中，可以根据应用特点进行多方面特性间的权衡研究。这些特性包括：网络传输效率、计算开销、存储开销、安全性、可靠性(密钥恢复)、网络带宽占用等。如多媒体数据的传输较普通数据需要更大的网络带宽，所以，在必要的情况下，需要牺牲部分安全性或增大计算开销或增大存储开销，以换取网络传输的高效性。这方面的研究将成为今后多媒体安全组通信中密钥管理研究的重要内容。

(2) 结合具体的应用，进行组密钥传输方式的研究。如采用媒体相关信道传输方式，将密钥生成、分发、更新的消息包嵌入视频媒体数据流进行传输，以减小密钥相关消息数据被嗅探、截获的可能性，提高系统的安全性，提高消息传递和密钥更新效率。

(3) 针对小规模、高安全性需求的多媒体组通信应用(如安全视频会议、安全电子白板)进行密钥协商方案的研究；针对组成员相对独立、广泛分布、成员关系动态变化的多媒体组通信应用，进行大规模、分布式安全组通信密钥管理方案的研究，进而推广到 P2P 应用模式中。

4.7 无线传感器网络中的密钥管理

4.7.1 无线传感器网络概述

普遍网络化孕育的传感器网络是一种新的信息获取和处理技术。在特殊领域,它有着传统技术不可比拟的优势,同时也必将开辟出不少新颖而有价值的商业应用。我们归纳和总结了已有的研究,着重介绍路由和介质访问控制等与网络密切相关的技术问题,并对一些可能的研究方向进行了简要的阐述[83]。

4.7.1.1 传感器网络的特点

更小、更廉价的低功耗计算设备代表的“后 PC 时代”冲破了传统台式计算机和高性能服务器的设计模式;普遍的网络化带来的计算处理能力是难以估量的;微机电系统(MEMS)的迅速发展奠定了设计和实现片上系统(SoC)的基础。上述 3 方面的高度集成又孕育出了许多新的信息获取和处理模式,传感器网络就是其中一例。

随机分布的集成由传感器、数据处理单元和通信模块的微小节点通过自组织的方式构成网络,借助于节点中内置的形式多样的传感器测量所在周边环境中的热、红外、声呐、雷达和地震波信号,从而探测包括温度、湿度、噪声、光强度、压力、土壤成分、移动物体的大小、速度和方向等众多我们感兴趣的物质现象。在通信方式上,虽然可以采用有线、无线、红外和光等多种形式,但一般认为短距离的无线低功率通信技术最适合传感器网络使用,为明确起见,一般称作无线传感器网络。但也不绝对,Berkeley 的 Smart Dust[84] 因为可以像尘埃一样悬浮在空中,有效地避免了障碍物的遮挡,因此采用光作为通信介质。

无线传感器网络与传统的无线网络(如 WLAN 和蜂窝移动电话网络)有着不同的设计目标,后者在高度移动的环境中通过优化路由和资源管理策略最大化带宽的利用率,同时为用户提供一定的服务质量保证。在无线传感器网络中,除了少数节点需要移动以外,大部分节点都是静止的。因为它们通常运行在人无法接近的恶劣甚至危险的远程环境中,能源无法替代,设计有效的策略延长网络的生命周期成为无线传感器网络的核心问题。当然,从理论上讲,太阳能电池能够持久地补给能源,但工程实践中生产这种微型化的电池还有相当的难度。在无线传感器网络的研究初期,人们一度认为成熟的 Internet 技术加上 Ad-hoc 路由机制对传感器网络的设计是足够充分的,但更深入的研究表明:传感器网络有着与传统网络明显不同的技术要求。前者以数据为中心,后者以传输数据为目的。为了适应广泛的应用程序,传统网络的设计遵循着端到端边缘论思想[85],强调将一切与功能相关的处理都放在网络的端系统上,中间节点仅仅负责数据分组的转发,对于传感器网络,这未必是一种合理的选择。一些为自组织的 Ad-hoc 网络设计的协议和算法未必适合传感器网络的特点和应用的要求。节点标识(如地址等)的作用在传感器网络中就显得不是十分重要,因为应用程序不关心单节点上的信息;中间节点上与具体应用相关的数据处理、融合和缓存也显得很有必要。在密集型的传感器网络中,相邻节点间的距离非常短,低功耗的多跳通信模式节省功耗,同时增加了通信的隐蔽性,也避免了长距离的无线通信易受外界噪声干扰的影响。这

些独特的要求和制约因素为传感器网络的研究提出了新的技术问题。

4.7.1.2　传感器网络的体系结构

1. **节点组成**

在不同的应用中，传感器网络节点的组成不尽相同，但一般都由数据采集、数据处理、数据传输和电源这 4 部分组成。被监测物理信号的形式决定了传感器的类型。处理器通常选用嵌入式 CPU，如 Motorola 的 68HC16、ARM 公司的 ARM7 和 Intel 的 8086 等。数据传输单元主要由低功耗、短距离的无线通信模块组成，比如 RFM 公司的 TR1000 等。因为需要进行较复杂的任务调度与管理，因此系统需要一个微型化的操作系统，UC Berkeley 为此专门开发了 TinyOS[86]，当然，uCOS-Ⅱ和嵌入式 Linux 等也是不错的选择。图 4.7.1 描述了节点的组成，其中实心箭头的方向表示数据在节点中的流动方向。

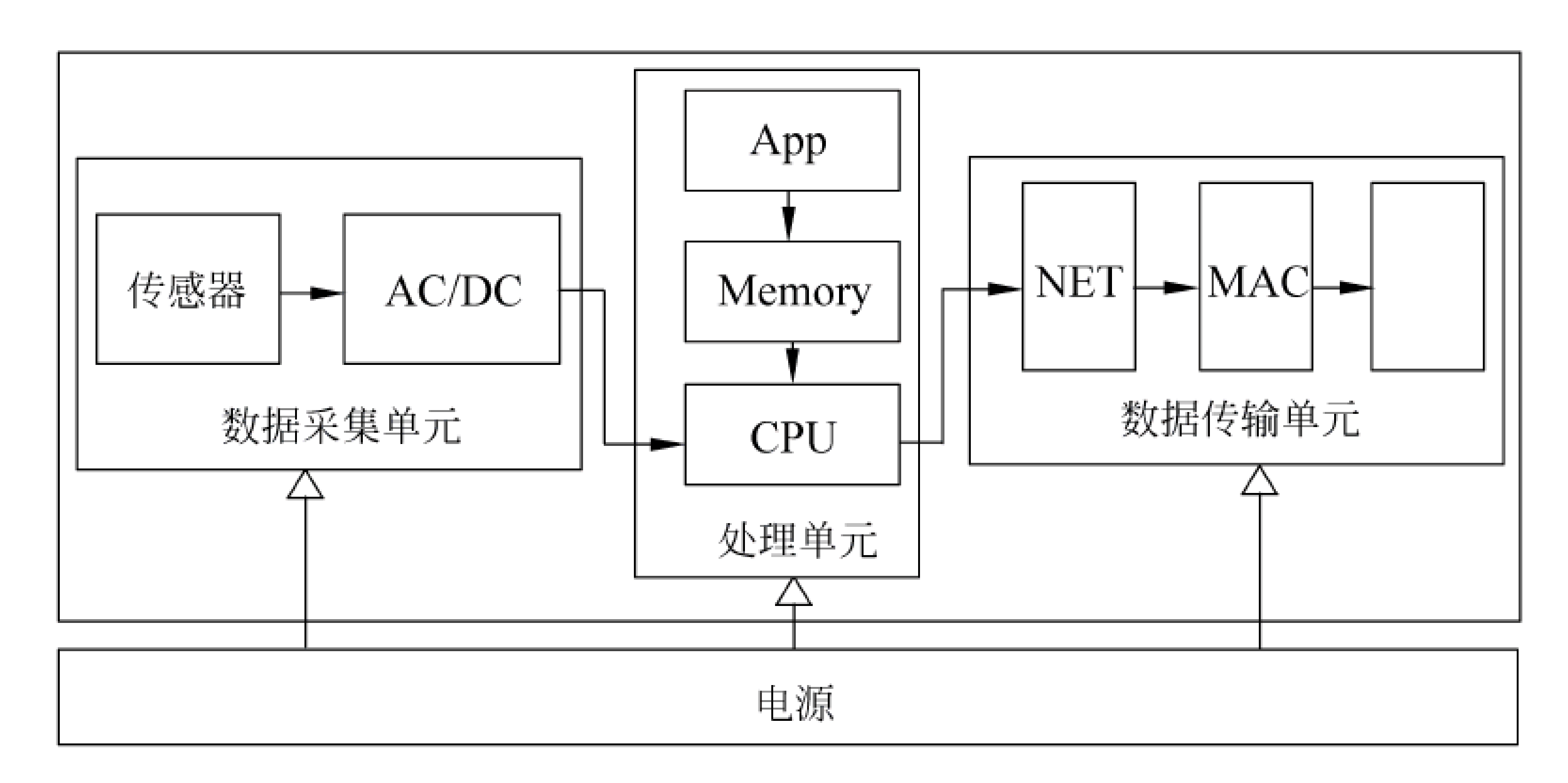


图 4.7.1　传感器网络节点的组成

2. **网络体系结构**

在传感器网络中，节点任意散落在被监测区域内，这一过程是通过飞行器撒播、人工埋置和火箭弹射等方式完成的。节点以自组织形式构成网络，通过多跳中继方式将监测数据传到 sink 节点，最终借助长距离或临时建立的 sink 链路将整个区域内的数据传送到远程中心进行集中处理。卫星链路可用作 sink 链路，借助游弋在监测区上空的无人飞机回收 sink 节点上的数据也是一种方式，UC Berkeley 在进行 UAV(unmanned aerial vehicle)项目[87]的外场测试时便采用了这种方式。如果网络规模太大，可以采用聚类分层的管理模式，图 4.7.2 给出了传感器网络体系结构一般形式的描述。

4.7.1.3　传感器网络的应用

MEMS 支持下的微小传感器技术和节点间的无线通信能力为传感器网络赋予了广阔的应用前景，主要表现在军事、环境、健康、家庭和其他商业领域。当然，在空间探索和灾难拯救等特殊领域，传感器网络也有其得天独厚的技术优势。

1. **军事应用**

在军事领域，传感器网络将会成为 C4ISRT(command，control，communication，computing，intelligence，surveillance，reconnaissance and targeting)系统不可或缺的一部分。

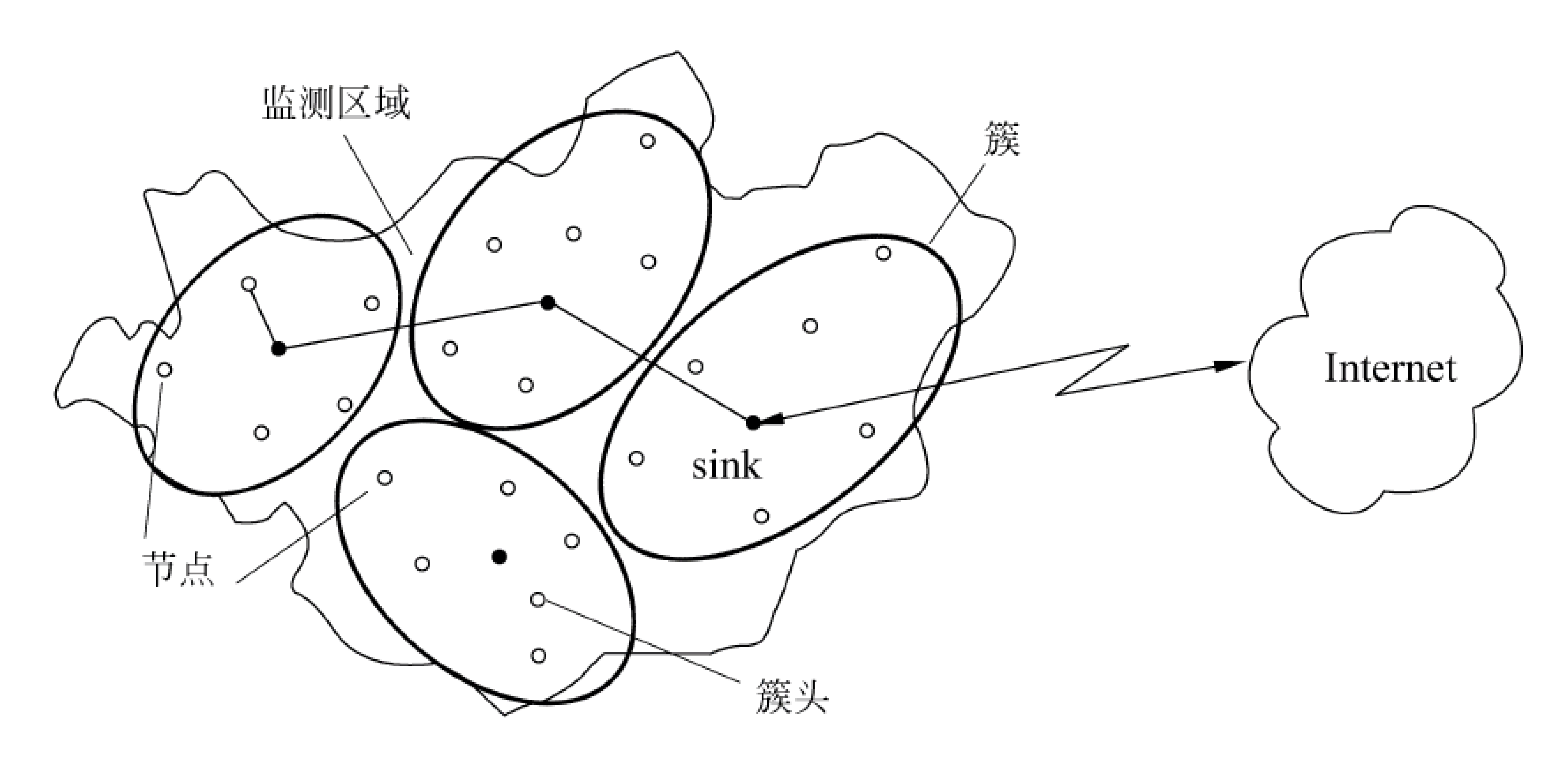


图 4.7.2 传感器网络的体系结构

C4ISRT 系统的目标是利用先进的高科技为未来的现代化战争设计一个集命令、控制、通信、计算、智能、监视、侦察和定位于一体的战场指挥系统，受到了军事发达国家的普遍重视。因为传感器网络是由密集型、低成本、随机分布的节点组成的，自组织性和容错能力使其不会因为某些节点在恶意攻击中的损坏而导致整个系统的崩溃，这一点是传统的传感器技术所无法比拟的，也正是这一点，使传感器网络非常适合应用于恶劣的战场环境中，包括监控我军兵力、装备和物资，监视冲突区，侦察敌方地形和布防，定位攻击目标，评估损失，侦察和探测核、生物和化学攻击。在战场，指挥员往往需要及时、准确地了解部队、武器装备和军用物资供给的情况，铺设的传感器将采集相应的信息，并通过汇聚节点将数据送至指挥所，再转发到指挥部，最后融合来自各战场的数据形成完备的战区态势图。在战争中，对冲突区和军事要地的监视也是至关重要的，通过铺设传感器网络，以更隐蔽的方式近距离地观察敌方的布防；当然，也可以直接将传感器节点撒向敌方阵地，在敌方还未来得及反应时迅速收集利于作战的信息。传感器网络也可以为火控和制导系统提供准确的目标定位信息。在生物和化学战中，利用传感器网络及时、准确地探测爆炸中心将会为我军提供宝贵的反应时间，从而最大可能地减小伤亡。传感器网络也可以避免核反应部队直接暴露在核辐射的环境中。在军事应用中，与独立的卫星和地面雷达系统相比，传感器网络的潜在优势表现在以下几个方面：

(1) 分布节点中多角度和多方位信息的综合有效地提高了信噪比，这一直是卫星和雷达这一类独立系统难以克服的技术问题之一。

(2) 传感器网络低成本、高冗余的设计原则为整个系统提供了较强的容错能力。

(3) 传感器节点与探测目标的近距离接触极大地消除了环境噪声对系统性能的影响。

(4) 节点中多种传感器的混合应用有利于提高探测的性能指标。

(5) 多节点联合，形成覆盖面积较大的实时探测区域。

(6) 借助于个别具有移动能力的节点对网络拓扑结构的调整能力，可以有效地消除探测区域内的阴影和盲点。

2. 环境科学

随着人们对于环境的日益关注，环境科学所涉及的范围越来越广泛。通过传统方式采集原始数据是一件困难的工作。传感器网络为野外随机性的研究数据获取提供了方便，比

如，跟踪候鸟和昆虫的迁移，研究环境变化对农作物的影响，监测海洋、大气和土壤的成分等。ALERT[88]系统中就有数种传感器用于监测降雨量、河水水位和土壤水分，并依此预测爆发山洪的可能性[89]。类似地，传感器网络对森林火灾准确、及时地预报也是有帮助的。此外，传感器网络也可以应用在精细农业中，以监测农作物中的害虫、土壤的酸碱度和施肥状况等。

3. 医疗健康

如果在住院病人身上安装特殊用途的传感器节点，如心率和血压监测设备，利用传感器网络，医生就可以随时了解被监护病人的病情，进行及时处理[90]。还可以利用传感器网络长时间地收集人的生理数据，这些数据在研制新药品的过程中是非常有用的，而安装在被监测对象身上的微型传感器也不会给人的正常生活带来太多的不便。此外，在药物管理等诸多方面，也有新颖而独特的应用。总之，传感器网络为未来的远程医疗提供了更加方便、快捷的技术实现手段。

4. 空间探索

探索外部星球一直是人类梦寐以求的理想，借助于航天器布撒的传感器网络节点实现对星球表面长时间的监测，应该是一种经济、可行的方案。NASA 的 JPL(Jet Propulsion Laboratory)实验室研制的 Sensor Webs[91]就是为将来的火星探测进行技术准备的，已在佛罗里达宇航中心周围的环境监测项目中进行测试和完善。

5. 其他商业应用

自组织、微型化和对外部世界的感知能力是传感器网络的三大特点，这些特点决定了传感器网络在商业领域应该也会有不少的机会。比如，嵌入家具和家电中的传感器与执行机构组成的无线网络与 Internet 连接在一起将会为我们提供更加舒适、方便和具有人性化的智能家居环境；文献[92]中描述的城市车辆监测和跟踪系统中成功地应用了传感器网络；德国某研究机构正在利用传感器网络技术为足球裁判研制一套辅助系统，以减小足球比赛中越位和进球的误判率。此外，在灾难拯救、仓库管理、交互式博物馆、交互式玩具、工厂自动化生产线等众多领域，无线传感器网络都将会孕育出全新的设计和应用模式。

4.7.1.4　传感器网络在网络层研究的热点问题

迄今为止，传感器网络的研究大致经过了两个阶段。第 1 阶段主要偏重利用 MEMS 技术设计小型化的节点设备，代表性的研究项目有 WINS[93]和 Smart Dust。对于网络本身问题的关注和研究可以认为是传感器网络研究的第 2 个阶段，目前正在成为无线网络研究领域的一个不小的热点。从网络分层模型的角度分析，每一层都有需要结合传感器网络的特点进行细致研究的问题，就已有的研究而言，主要集中在网络层和链路层。

传感器网络中的路由协议分为平面型和层次型两种，但大都采用多跳形式在节点和易移动的 sink 节点之间建立连接。Ad-hoc 网络中已有的多跳路由协议，如 AODV(ad-hoc demand distance vector)和 TORA(temporally ordered routing algorithm)等，一般都不适合传感器网络的特点和要求。传感器中的大部分节点不像 Ad-hoc 网络中的节点那样快速移动，因此没有必要花费很大的代价频繁地更新路由表信息。

1. 平面路由协议

(1) Flooding

泛洪是一种传统的路由技术,不要求维护网络的拓扑结构,并进行路由计算,接收到消息的节点以广播形式转发分组。对于自组织的传感器网络,泛洪路由是一种较直接的实现方法,但消息的"内爆(implosion)"和"重叠(overlap)"是其固有的缺陷。为了克服这些缺陷,S. Hedetniemi 等人提出了 Gossiping 策略[94],节点随机选取一个相邻节点转发它接收到的分组,而不是采用广播形式。这种方法避免了消息的"内爆"现象,但有可能增加端到端的传输延时。

(2) SPIN (sensor protocol for information via negotiation)[95]

SPIN 是以数据为中心的自适应路由协议,通过协商机制来解决泛洪算法中的"内爆"和"重叠"问题。传感器节点仅广播采集数据的描述信息,当有相应的请求时,才有目的地发送数据信息。SPIN 协议中有 3 种类型的消息,即 ADV,REQ 和 DATA。节点用 ADV 宣布有数据发送,用 REQ 请求希望接收数据,用 DATA 封装数据。SPIN 协议有 4 种不同的形式:

- SPIN-PP:采用点到点的通信模式,并假定两节点间的通信不受其他节点的干扰,分组不会丢失,功率没有任何限制。要发送数据的节点通过 ADV 向它的相邻节点广播消息,感兴趣的节点通过 REQ 发送请求,数据源向请求者发送数据。接收到数据的节点再向它的相邻节点广播 ADV 消息,如此重复,使所有节点都有机会接收到任何数据。
- SPIN-EC:在 SPIN-PP 的基础上考虑了节点的功耗,只有能够顺利完成所有任务且能量不低于设定阈值的节点才可参与数据交换。
- SPIN-BC:设计了广播信道,使所有在有效半径内的节点可以同时完成数据交换。为了防止产生重复的 REQ 请求,节点在听到 ADV 消息以后,设定一个随机定时器来控制 REQ 请求的发送,其他节点听到该请求,主动放弃请求权利。
- SPIN-RL:它是对 SPIN-BC 的完善,主要考虑如何恢复无线链路引入的分组差错与丢失。记录 ADV 消息的相关状态,如果在确定时间间隔内接收不到请求数据,则发送重传请求,重传请求的次数有一定的限制。

(3) SAR (sequential assignment routing)[96]

在选择路径时,有序分配路由(SAR)策略充分考虑了功耗、QoS 和分组优先权等特殊要求,采用局部路径恢复和多路径备份策略,避免节点或链路失败时进行路由重计算需要的过量计算开销。为了在每个节点与 sink 节点间生成多条路径,需要维护多个树结构,每个树以落在 sink 节点有效传输半径内的节点为根向外生长,枝干的选择需满足一定 QoS 要求并要有一定的能量储备。这一处理使大多数传感器节点可能同时属于多个树,可任选其一将采集数据回传到 sink 节点。

(4) 定向扩散(directed diffusion)[97]

定向扩散模型是 Estrin 等人专门为传感器网络设计的路由策略,与已有的路由算法有着截然不同的实现机制。节点用一组属性值来命名它所生成的数据,比如将地震波传感器生成的数据命名为 Type=seismic,id=12,timestamp=02.01.22/21:10:23,locate-on =75-80S/100-120E。Sink 节点发出的查询业务也用属性的组合表示,逐级扩散,最终遍历

全网，找到所有匹配的原始数据。有一个称为“梯度”的变量与整个业务请求的扩散过程相联系，反映了网络中间节点对匹配请求条件的数据源的近似判断。更直接的方法是，节点用一组标量值表示它的选择，值越大意味着向该方向继续搜索获得匹配数据的可能性越大，这样的处理最终将会在整个网络中为 sink 节点的请求建立一个临时的“梯度”场，匹配数据可以沿“梯度”最大的方向中继回 sink 节点。图 4.7.3 描述了定向扩散模型的工作原理。

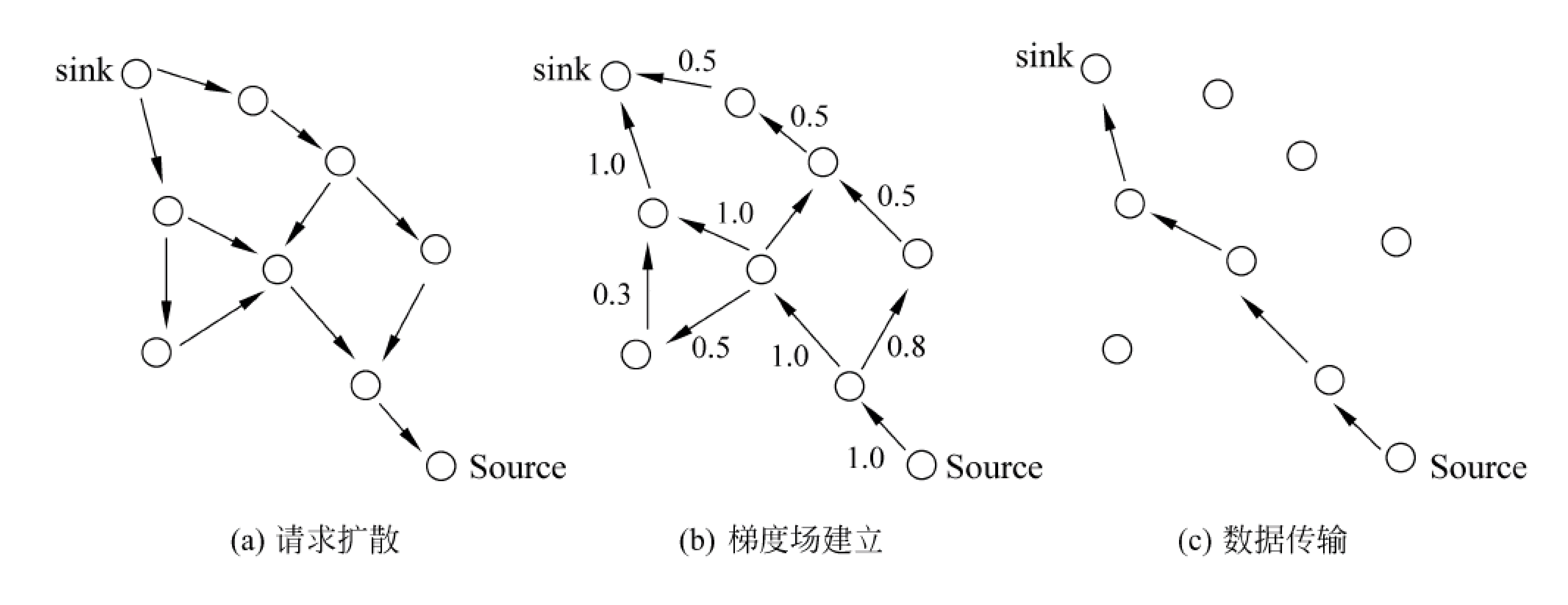


图 4.7.3 定向扩散路由原理

2. 层次路由协议

(1) LEACH (low energy adaptive clustering hierarchy)[98]

LEACH 是 MIT 的 Chandrakasan 等人为无线传感器网络设计的低功耗自适应聚类路由算法。与一般的平面多跳路由协议和静态聚类算法相比，LEACH 可以将网络生命周期延长 15%，主要通过随机选择聚类首领，平均分担中继通信业务来实现。LEACH 定义了“轮(round)”的概念，一轮由初始化和稳定工作两个阶段组成。为了避免额外的处理开销，稳定态一般持续相对较长的时间。

在初始化阶段，聚类首领是通过下面的机制产生的。传感器节点生成 0,1 之间的随机数，如果大于阈值 T，则选该节点为聚类首领。T 的计算方法如下：

$$T = \frac{p}{1 - p[r\text{mod}(1/p)]}$$

其中，p 为节点中成为聚类首领的百分数，r 是当前的轮数。一旦聚类首领被选定，它们便主动向所有节点广播这一消息。依据接收信号的强度，节点选择它所要加入的组，并告知相应的聚类首领。基于时分复用的方式，聚类首领为其中的每个成员分配通信时隙。在稳定工作阶段，节点持续采集监测数据，传与聚类首领，进行必要的融合处理之后，发送到 sink 节点，这是一种减小通信业务量的合理工作模式。持续一段时间以后，整个网络进入下一轮工作周期，重新选择聚类首领。

(2) TEEN (threshold sensitive energy efficient sensor network protocol)[99]

依照应用模式的不同，通常可以简单地将无线自组织网络(包括传感器网络和 Ad-hoc 网络)分为主动(proactive)和响应(reactive)两种类型。主动型传感器网络持续监测周围的物质现象，并以恒定速率发送监测数据；而响应型传感器网络只是在被观测变量发生突变时才传送数据。相比之下，响应型传感器网络更适用于敏感时间的应用中。TEEN 与 LEACH 的实现机制非常相似，只是前者是响应型的，而后者属于主动型传感器网络。在

TEEN 中定义了硬、软两个门限值，以确定是否需要发送监测数据。当监测数据第一次超过设定的硬门限时，节点用它作为新的硬门限，并在接着到来的时隙内发送它。在接下来的过程中，如果监测数据的变化幅度大于软门限界定的范围，则节点传送最新采集的数据，并将它设定为新的硬门限。通过调节软门限值的大小，可以在监测精度和系统能耗之间取得合理的平衡。NS2 平台上的仿真研究结果表明[100]：TEEN 比 LEACH 更有效。

（3）PEGASIS（power-efficient gathering in sensor information system）[101]

PEGASIS 由 LEACH 发展而来。它假定组成网络的传感器节点是同构且静止的。节点发送能量递减的测试信号，通过检测应答来确定离自己最近的相邻节点。通过这种方式，网络中的所有节点能够了解彼此的位置关系，进而每个节点依据自己的位置选择所属的聚类，聚类的首领参照位置关系优化出到 sink 节点的最佳链路。因为 PEGASIS 中每个节点都以最小功率发送数据分组，并有条件完成必要的数据融合，减小业务流量，因此，整个网络的功耗较小。研究结果表明，PEGASIS 支持的传感器网络的生命周期是 LEACH 的近两倍。PEGASIS 协议的不足之处在于节点维护位置信息（相当于传统网络中的拓扑信息）需要额外的资源。

（4）多层聚类算法[97]

多层聚类算法是 Estrin 为传感器网络设计的一种新的聚类实现机制。工作在网络中的传感器节点处于不同的层，所处层次越高，则所覆盖面积越大。起初，所有节点均在最低层，通过竞争获得提升高层的机会。当新的工作周期开始时，每一个节点都广播自己的状态信息，包括储备能量、所在层次和首领的 ID（如果有）等，然后进入等待状态以便相互了解信息，等待时间与所在层次成正比。处在最底层的节点如果没有首领，在等待状态结束后，立刻启动一个“晋升定时器”，定时时间与自身能量以及接收到同层其他节点广播消息的数目成反比，目的是为能量较高且在密集区的节点获得较多的提升机会。一旦定时时间到，节点升入高层，将有发给自己广播消息的节点视为潜在的子节点，并广播自己新的状态信息，低层节点选择响应这些准首领的广播消息，最终确定唯一的通信关系。选择了首领的节点，自己的“晋升定时器”将停止工作，也就意味着本轮放弃了晋升机会。在每一个工作周期结束以后，高层节点将视自己的状态信息（如有无子节点，功率是否充足）来决定是否让出首领位置。上述多层聚类算法具有递归性，Estrin 等人用两层模型验证了它在传感器网络中的有效性。

4.7.1.5 传感器网络在链路层研究的热点问题

链路层协议用于建立可靠的点到点或点到多点通信链路，主要由介质访问控制（MAC）组成。就实现机制而言，MAC 协议分为 3 类：确定性分配、竞争占用和随机访问。前两者不是传感器网络的理想选择。因为 TDMA 固定时隙的发送模式功耗过大，为了节省功耗，空闲状态应关闭发射机；竞争占用方案需要实时监测信道状态，也不是一种合理的选择；随机介质访问模式比较适合于无线传感网络的节能要求。

蜂窝电话网络、Ad-hoc 和蓝牙技术是当前主流的无线网络技术，但它们各自的 MAC 协议不适合无线传感器网络。GSM 和 CDMA 中的介质访问控制主要关心如何满足用户的 QoS 要求和节省带宽资源，功耗是第二位的；Ad-hoc 网络则考虑如何在节点具有高度移动性的环境中建立彼此间的链接，同时兼顾一定的 QoS 要求，功耗也不是其首要关心的；而蓝

牙采用了主从式的星形拓扑结构，这本身就不适合传感器网络自组织的特点。

基于以上两个方面的原因，需要为传感器网络设计新的低功耗 MAC 协议。下面我们简单介绍几种已有的典型方案。

1. SMACS[102]

SMACS 是分布式的 MAC 协议，无需任何局部或全局主节点的调度便能让传感器节点发现相邻节点，并安排合理信道占用时间。在具体实现中，相邻节点的发现和信道的分配是一起完成的，因此，当节点听到它所有的相邻节点时，也就意味着已经建立相应的通信子网，链路由固定频率、随机选择的时隙组成。SMACS 无需全网的时间同步机制，但在各子网内部保持同步是必要的。在竞争信道资源时，带延时的随机唤醒机制有效地减小了能量的损耗。SMACS 的缺点是时隙分配方案不够严密，属于不同子网的节点之间有可能永远得不到通信机会。

2. 基于 CSMA 的介质访问控制[103]

传统的载波侦听/多路访问(CSMA)机制不适合传感器网络的原因有两个：其一，持续侦听信道的过量功耗；其二，倾向支持独立的点到点通信业务，这样容易导致临近网关的节点获得更多的通信机会，而抑制多跳业务流量，造成不公平。为了弥补这些缺陷，Woo 和 Culler 从两个方面对传统的 CSMA 进行了改进，以适应传感器网络的技术要求：①采用固定时间间隔的周期性侦听方案节省功耗；②设计自适应传输速率控制(adaptive transmission rate control，ARC)策略，有针对性地抑制单跳通信业务量，为中继业务提供更多的服务机会，提高公平性。相似的工作还有 Ye 等人设计的 SMAC(sensor media access control)协议[104]。它也是利用周期性侦听机制节省功耗，但没有考虑公平性问题，而是在 PAMAS(power aware multi-access protocol with signalling)[105]的启发下，精简了用于同步和避免冲突的信令机制。以上两种基于 CSMA 改进的传感器网络 MAC 协议都在 TinyOS 微操作系统上进行了实现，并分别在 SmartDust[87]硬件平台上进行了测试，比 802.11 标准定义的 MAC 协议节省了 1～5 倍的功耗，基本上可为传感器网络所用。

3. TDMA/FDMA 组合方案[106]

Sohrabi 和 Pottie 设计的传感器网络自组织 MAC 协议是一种时分复用和频分复用的混合方案，具有一定的代表性。节点上维护着一个特殊的结构帧，类似于 TDMA 中的时隙分配表，节点据此调度它与相邻节点间的通信。FDMA 技术提供的多信道，使多个节点之间可以同时通信，有效地避免了冲突。只是在业务量较小的传感器网络中，该组合协议的信道利用率较低，因为事先定义的信道和时隙分配方案限制了对空闲时隙的有效利用。

4.7.1.6 其他重要的热点问题

除了网络自身的问题以外，还有许多关键问题也引起了研究者广泛的兴趣，主要集中在两个方面，即如何从系统角度出发节省功耗以及与应用相关的共性技术。

1. 系统节能策略

(1) 动态功率管理[107]

在多数传感器网络的应用中，监测事件具有很强的偶发性，节点上所有的工作单元没有必要时刻保持在正常的工作状态。处于沉寂状态，甚至完全关闭，必要时加以唤醒是一种有

效的系统节能方案。传感器网络节点的主要功耗器件有处理器、内存、带 A/D 的传感器和无线收发单元。Sinhua 等人根据它们的状态组合的有效性,将整个节点分为 5 种工作状态,在嵌入式操作系统的支持下进行切换,既满足了功能的需要,又节省了功耗。

(2) 动态电压调度

在文献[108]中,由 Lm 等人提出的动态电压调度(dynamic voltage scheduling, DVS)策略的主要原理是基于负载状态动态调节供电电压来减小系统功耗,并被应用到 PDA 之类的个人移动设备上。由此受到启发,我们将其应用到传感器网络中,提出了如图 4.7.4 所示的功率控制原理图。节点上的嵌入式操作系统负责调度来自不同任务队列的请求接受服务,并实时监测处理器的利用率和任务队列的长度,负载观测器依据这两个参数的序列值计算负载的标称值 w,直流/直流变换器参照该值输出幅值为 A 的电压,支持处理器的正常工作。这构成了一个典型的闭环反馈系统。控制理论中成熟的方法可以为该系统中各个模块的设计提供有力的支持。

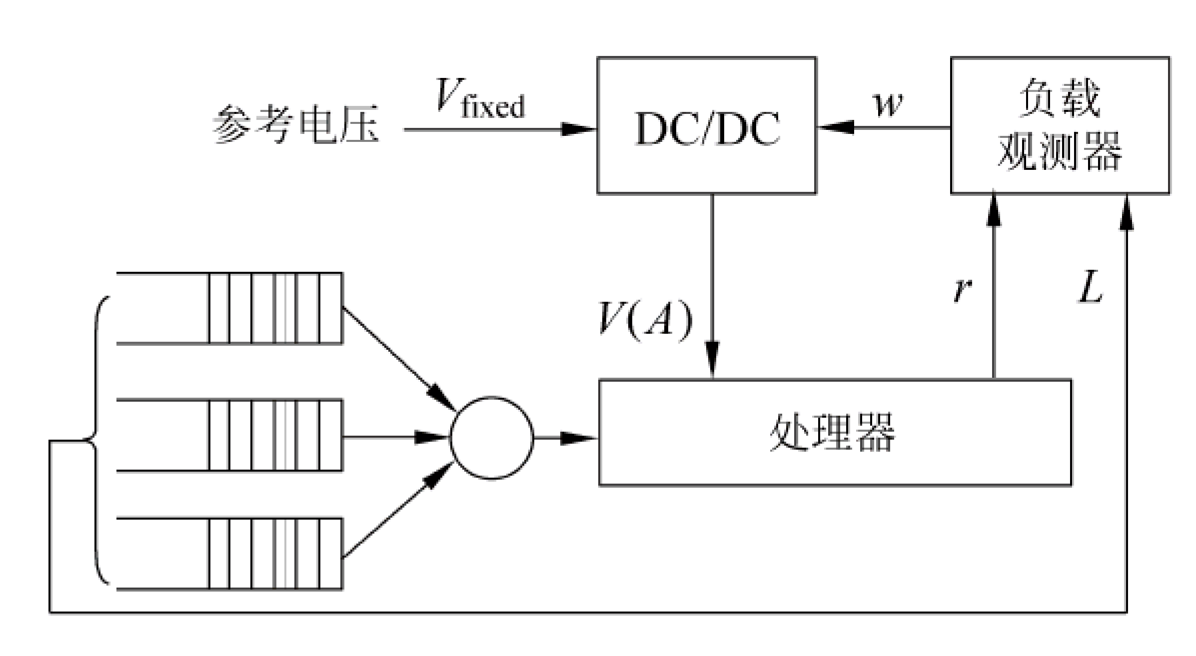


图 4.7.4 DVS 功率控制原理图

2. 共性技术

在大多数传感器网络的应用中,诸如目标定位和时间同步等一些共性技术的支持是必不可少的,在军事应用中它们显得更为重要,因此,吸引了不少研究者的注意。

(1) 时钟同步

传感器网络中的通信协议和应用,比如基于 TDMA 的 MAC 协议和敏感时间的监测任务等,要求节点间的时钟必须保持同步。在文献[109]中,Elson 和 Estrin 给出了一种简单、实用的同步策略。其基本思想是,节点以自己的时钟记录事件,随后用第三方广播的基准时间加以校正,精度依赖于对这段间隔时间的测量。这种同步机制应用在确定来自不同节点的监测事件的先后关系时有足够的精度。设计高精度的时钟同步机制是传感网络设计和应用中的一个技术难点。我们认为,考虑精简 NTP(network time protocol)协议的实现复杂度,将其移植到传感器网络中应该是一个有价值的研究课题。

(2) 定位机制与算法

定位是大多数应用,特别是军事应用的基础。传感器网络中的定位机制与算法包括两部分:节点自身定位和外部目标定位,前者是后者的基础[110]。在节点自身定位方面,DARPA 支持的一些有军事应用背景的项目,如 DSN(dynamic sensor network)[111] 和 SCADDS(scalable coordination architecture for deeply distributed and dynamic system)[112] 等,大多采用 GPS(global positioning system)技术。对于一些定位精度要求不高的项目,则

应用了 LPS(local positioning system)[113]。由于 GPS 不适合中国的军事国情,我们设想了一种依赖于自己技术实现传感器网络中节点定位的机制,如图 4.7.5 所示。在"北斗一号"双星定位系统的支持下,传感器网络中的某些节点就可以找到自己的精确位置,然后参照此基准,利用局部定位算法,其他节点也可以正确定位。此外,在这种模式下,"北斗一号"的上行数据通路恰好可以作为传感器网络的 sink 链路,将数据回传给控制中心,省去了用飞行器等其他手段收集数据的麻烦。确定了节点的基准位置,利用传统的定位机制和算法,如接收信号的强弱、角度和时间等,以及典型的三角形算法,就可以定位外部目标,这是相对成熟的技术。

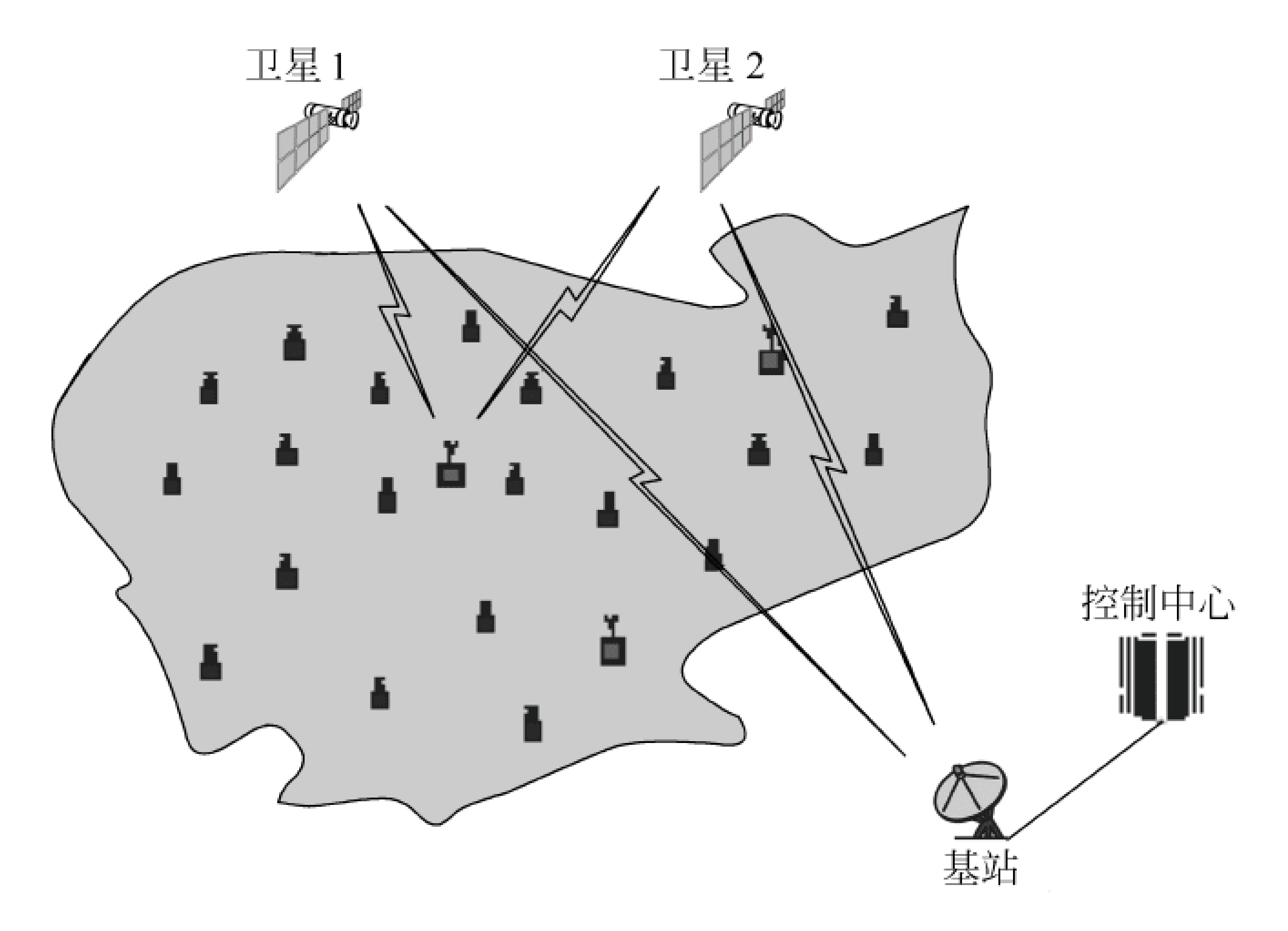


图 4.7.5 传感器节点定位系统原理图

4.7.2 无线传感器网络密钥管理研究现状

无线传感器网络 WSN 集微机电技术、传感器技术、通信技术于一体,可广泛应用于教育、军事、医疗、交通等诸多领域,拥有巨大的应用潜力和商业价值[114,115],引起了国内外广泛的关注和研究[116~119]。安全是 WSN 最基本的一项服务,特别是当 WSN 被部署在无人触及或容易受损或被俘获的环境时,保证 WSN 的安全性更是应予以优先考虑的问题[120,121]。以提供安全、可靠的保密通信为目标的密钥管理是 WSN 安全研究最为重要、最为基本的内容,有效的密钥管理机制也是其他安全机制,如安全路由[122]、安全定位[123]、安全数据融合[124]及针对特定攻击的解决方案[125]等的基础。

在传统网络中,密钥管理的研究与应用中已取得许多成果[126~128]。但是因为 WSN 所固有的特点,使得这些研究成果一般不能直接应用于 WSN。具体表现在:①WSN 节点资源(包括存储容量、计算能力、通信带宽和距离等)受到更加严格的限制。例如,UCB(University of California at Berkeley)研制的 MICA2 mote[129],使用 8 位 7.3828 MHz ATmega 128L 处理器,SRAM 为 4 KB,ROM 为 128 KB,通信频率为 916 MHz,带宽为 10 Kbps。资源的严格受限使得传统的对节点计算、存储和通信开销较大的密钥管理方案或协议无法应用于 WSN。②一般而言,WSN 没有固定的基础设施支持。因此,基于在线的密钥分配中心(key distribution center,KDC)的密钥管理方案或协议[126]无法应用于 WSN。

③节点容易受损。WSN 节点一般被设计为无特殊物理保护的、容易受到物理损坏或被俘获，网络中的部分节点处于非正常运行状态是一个普遍现象，一些状态敏感的密钥管理方案或协议[127,128]就无法应用于 WSN。

4.7.3 无线传感器网络密钥管理的安全和性能评价

与典型网络一样，WSN 密钥管理必须满足可用性（availability）、完整性（integrity）、机密性（confidentiality）、认证（authentication）和不可否认（non-reputation）等传统的安全需求[130,131]。此外，根据 WSN 自身的特点，WSN 密钥管理还应满足如下一些性能评价指标：

（1）可扩展性（scalability）。WSN 的节点规模少则十几个或几十个，多则成千上万。随着规模的扩大，密钥协商所需的计算、存储和通信开销都会随之增大，密钥管理方案和协议必须能够适应不同规模的 WSN。

（2）有效性（efficiency）。网络节点的存储、处理和通信能力非常受限的情况必须充分考虑。具体而言，应考虑以下几个方面：存储复杂度（storage complexity），用于保存通信密钥的存储空间使用情况；计算复杂度（computation complexity），为生成通信密钥而必须进行的计算量情况；通信复杂度（communication complexity），在通信密钥生成过程中需要传送的信息量情况。

（3）密钥连接性（key connectivity）。节点之间直接建立通信密钥的概率。保持足够高的密钥连接概率是 WSN 发挥其应有功能的必要条件。需要强调的是，WSN 节点几乎不可能与距离较远的其他节点直接通信，因此并不需要保证某一节点与其他所有的节点保持安全连接，仅需确保相邻节点之间保持较高的密钥连接。

（4）抗毁性（resilience）。抵御节点受损的能力。也就是说，存储在节点的或在链路交换的信息未给其他链路暴露任何安全方面的信息。抗毁性可表示为当部分节点受损后，未受损节点的密钥被暴露的概率。抗毁性越好，意味着链路受损就越低。

4.7.4 无线传感器网络密钥管理方案和协议的分类

近年来，WSN 密钥管理的研究已经取得许多进展[132]。不同的方案和协议，其侧重点也有所不同。下面我们依据这些方案和协议的特点进行适当的分类。

1. 对称密钥管理与非对称密钥管理

根据所使用的密码体制，WSN 密钥管理可分为对称密钥管理和非对称密钥管理两类。在对称密钥管理方面，通信双方使用相同的密钥和加密算法对数据进行加密、解密，对称密钥管理具有密钥长度不长，计算、通信和存储开销相对较小等特点，比较适用于 WSN，目前是 WSN 密钥管理的主流研究方向。在非对称密钥管理方面，节点拥有不同的加密和解密密钥，一般都使用在计算意义上安全的加密算法。非对称密钥管理由于对节点的计算、存储、通信等能力要求比较高，曾一度被认为不适用于 WSN，但一些研究[133,134]表明，非对称加密算法经过优化后能够适用于 WSN。从安全的角度来看，非对称密码体制的安全强度在计算意义上要远远高于对称密码体制。

2. 分布式密钥管理和层次式密钥管理

根据网络的结构，WSN 密钥管理可分为分布式密钥管理和层次式密钥管理两类。在分布式密钥管理[135~143]中，节点具有相同的通信能力和计算能力。节点密钥的协商、更新通过使用节点预分配的密钥和相互协作来完成。而在层次 WSN 密钥管理[144~147]里，节点被划分为若干簇，每一簇有一个能力较强的簇头（cluster head）。普通节点的密钥分配、协商、更新等都是通过簇头来完成的。

分布式密钥管理的特点是密钥协商通过相邻节点的相互协作来实现，具有较好的分布特性。层次式密钥管理的特点是对普通节点的计算、存储能力要求低，但簇头的受损将导致严重的安全威胁。

3. 静态密钥管理与动态密钥管理

根据节点在部署之后密钥是否更新，WSN 密钥管理可分为静态密钥管理和动态密钥管理两类[148]。在静态密钥管理中，节点在部署前预分配一定数量的密钥，部署后通过协商生成通信密钥，通信密钥在整个网络运行期内不考虑密钥更新和撤回；而在动态密钥管理中，密钥的分配、协商、撤回操作周期性地进行。

静态密钥管理的特点是通信密钥无需频繁更新，不会导致更多的计算和通信开销，但不排除受损节点继续参与网络操作。若存在受损节点，则对网络具有安全威胁。动态密钥管理的特点是可以使节点通信密钥处于动态更新状态，攻击者很难通过俘获节点来获取实时的密钥信息，但密钥的动态分配、协商、更新和撤回操作将导致较大的通信和计算开销。

4. 随机密钥管理与确定密钥管理

根据节点的密钥分配方法区分，WSN 密钥管理可分为随机密钥管理与确定密钥管理两种。在随机密钥管理中，节点的密钥环通过随机方式获取，比如从一个大密钥池里随机选取一部分密钥[135]，或从多个密钥空间里随机选取若干个[137]。而在确定性密钥管理中，密钥环是以确定的方式获取的，比如，使用地理信息[138]，或使用对称 BIBD（balanced incomplete block design）[143]、对称多项式[149]等。从连通概率的角度来看，随机密钥管理的密钥连通概率介于 0 和 1 之间，而确定密钥管理的连通概率总为 1。

随机性密钥管理的优点是密钥分配简便，节点的部署方式不受限制；其缺点是，密钥的分配具有盲目性，节点可能存储一些无用的密钥而浪费存储空间。确定性密钥管理的优点是密钥的分配具有较强的针对性，节点的存储空间利用得较好，任意两个节点可以直接建立通信密钥；其缺点是，特殊的部署方式会降低灵活性，或密钥协商的计算和通信开销较大。

4.7.5 典型的无线传感器网络密钥管理的方案和协议

4.7.5.1 Eschenauer 随机密钥预分配方案[135]

Eschenauer 和 Gligor 在 WSN 中最先提出随机密钥预分配方案（简称 E-G 方案）。该方案由 3 个阶段组成。第 1 阶段为密钥预分配阶段。部署前，部署服务器首先生成一个密钥总数为 P 的大密钥池及密钥标识，每一节点从密钥池里随机选取 $k(k \ll P)$个不同密钥，这种随机预分配方式使得任意两个节点能够以一定的概率存在着共享密钥。第 2 阶段为共

享密钥发现阶段。随机部署后，两个相邻节点若存在共享密钥，就随机选取其中的一个作为双方的配对密钥(pair-wise key)；否则，进入到第 3 阶段。第 3 阶段为密钥路径建立阶段，节点通过与其他存在共享密钥的邻居节点经过若干跳后建立双方的一条密钥路径。

根据经典的随机图理论[150]，节点的度 d 与网络节点总数 n 存在以下关系：

$$d=\frac{n-1}{n}(\ln n-\ln(-\ln P_c)),$$

其中，P_c 为全网连通概率。若节点的期望邻居节点数为 $n'(n'\ll n)$，则两个相邻节点共享一个密钥的概率 $p'=\frac{d}{n'-1}$。在给定 p' 的情况下，P 和 k 之间的关系可以表示如下：

$$p'=1-\frac{((P-k)!)^2}{(P-2k)!P!}$$

E-G 方案在以下 3 个方面满足和符合 WSN 的特点：一是节点仅存储少量密钥就可以使网络获得较高的安全连通概率，例如，要保证节点数为 10 000 的 WSN 几乎保持全连通，每个节点仅需从密钥总数为 100 000 的密钥池随机选取 250 个密钥即可满足要求；二是密钥预分配时不需要节点的任何先验信息(如节点的位置信息、连通关系等)；三是部署后节点间的密钥协商无需 sink 的参与，使得密钥管理具有良好的分布特性。

4.7.5.2 对 E-G 方案的几种改进

E-G 方案的密钥随机预分配思想为 WSN 密钥预分配策略提供了一种可行的思路，后续许多方案和协议都在此框架基础上发展而来。它们分别从共享密钥阈值、密钥池结构、密钥预分配策略、密钥路径建立方法等方面提高随机密钥预分配方案的性能。

1. *q*-composite 随机密钥预分配方案[136]

在 Chan 提出的 q-composite 随机密钥预分配方案(简称 q-composite 方案)中，节点从密钥总数为 $|S|$ 的密钥池里预随机选取 m 个不同的密钥，部署后两个相邻节点至少需要共享 q 个密钥才能直接建立配对密钥。若共享的密钥数为 $t(t\geqslant q)$，则可使用单向散列函数建立配对密钥 $K=\mathrm{hash}(k_1\parallel k_2\parallel\cdots\parallel k_t)$(密钥序列号事先约定)。

随着共享密钥阈值的增大，攻击者能够破坏安全链路的难度呈指数增大，但同时对节点的存储空间需求也增大。因此，阈值 q 的选取是该方案需要着重考虑的一个因素。实验表明，当网络中的受损节点数量较少时，该方案的抗毁性比 E-G 方案要好，但随着受损节点数量的增多，该方案变得比较差。

2. 多密钥空间随机密钥预分配方案[137]

Blom 单密钥空间方案[151]使得网络中的任意两个节点都能够直接建立配对密钥，并且确保在受损节点数不超过阈值时，网络不会泄露任何机密信息。Du 将其扩展为多密钥空间随机密钥预分配方案[137]。网络节点总数为 N，部署前，部署服务器在有限域 $\mathrm{GF}(q)$(q 为足够大的素数)上生成一个 $(\lambda+1)\times N$ 的公开矩阵 G(G 满足任意 $\lambda+1$ 列线性不相关)和 ω 个 $(\lambda+1)\times(\lambda+1)$ 的对称机密矩阵 $D_1,D_2,\cdots,D_\omega$，每一对 $(D_i,G)_{i=1,2,\cdots,\omega}$ 称为一个密钥空间。部署服务器分别计算 $A_i=(D_i\times G)^{\mathrm{T}}$。每一节点随机选取 τ 个 $(2\leqslant\tau<\omega)$ 密钥空间，对于被节点 j 选中的矩阵 D_i，j 保存矩阵 A_i 的第 j 行元素，这些行元素信息是机密的，不公开，节点同时也保存矩阵 G 第 j 列相应的种子值(仅保留种子值是出于节约存储空间的考虑)。部

署后，若任意两个相邻节点共享一个密钥空间，就可以利用矩阵 A_i 的对称性直接建立配对密钥。配对密钥的生成如图 4.7.6 所示。

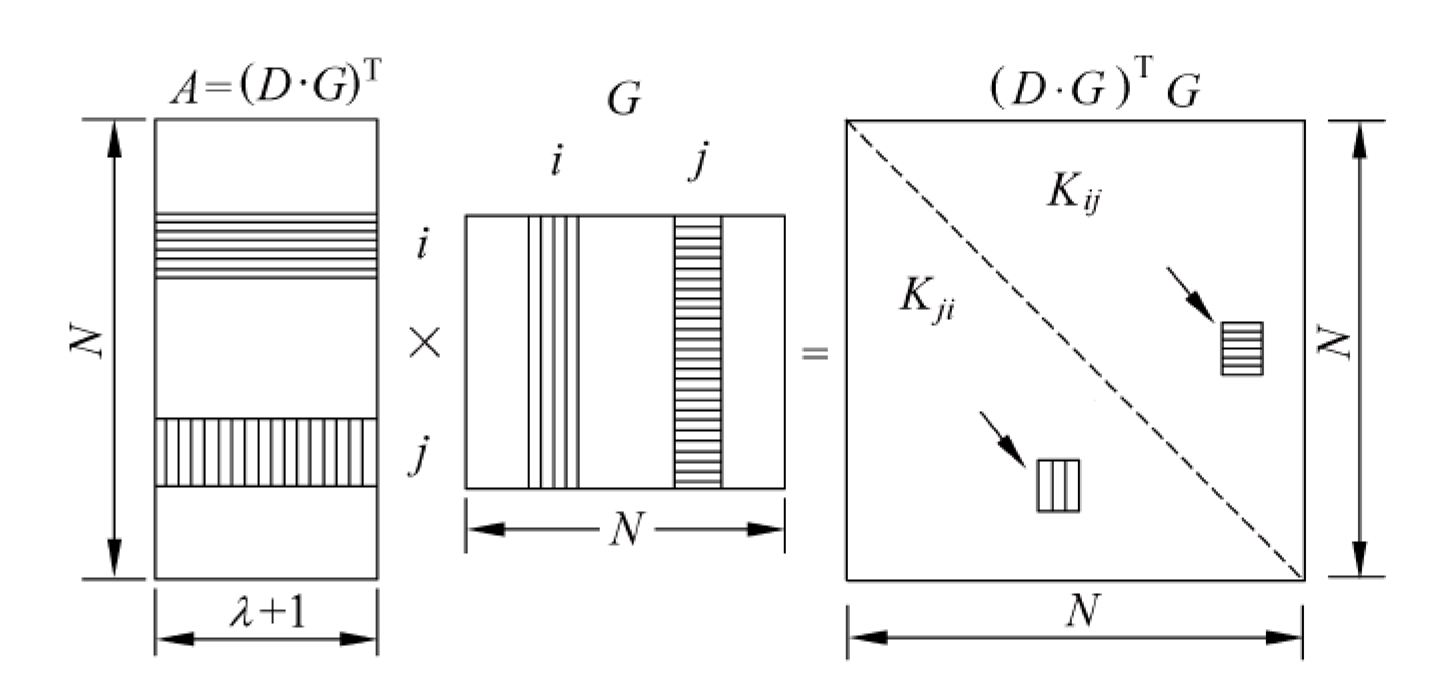


图 4.7.6　生成配对密钥

只要选择合适的 ω 和 τ 就能够提高密钥空间不被暴露的概率。实验表明，要使 10% 的安全链路受损，E-G 方案和 q-composite 方案就必须俘获比该方案 5 倍多数量的节点。方案的缺点是计算开销较大。与 Blom 方案相比，该方案虽然降低了密钥连通概率，但却提高了网络密钥连通的抗毁性。

3. 对称多项式随机密钥预分配方案[138]

Blundo 方案[149]使用对称二元多项式的性质$\left(f(x,y)=\sum_{i,j=0}^{t} a_{ij}x^i y^j \text{ 且 } f(x,y)=f(y,x)\right)$为网络中的任意两个节点建立配对密钥。Liu 在此基础上提出了基于多个对称二元多项式的随机密钥预分配方案[138]。部署前，部署服务器在有限域 F_q 上随机生成 s 个 t 阶对称二元多项式$\{f_i(x,y)\}_{i=1,2,\cdots,s}$；然后，每一节点随机选取 s' 个多项式共享。部署后，相邻节点若有相同的多项式共享，则直接建立配对密钥。

实验表明，当受损节点数较少时，该方案的抗毁性比 E-G 方案和 q-composite 方案要好，但当受损节点超过一定阈值时（如 60% 节点受损），该方案的安全链路受损数量则超过上述两个方案。

4. 基于地理信息或部署信息的随机密钥预分配方案[139~141]

在一些特殊的应用中，节点的位置信息或部署信息可以预先大概估计并用于密钥管理。Liu 在静态 WSN 里建立了基于地理信息的最靠近配对密钥（closest pairwise keys scheme，CPKS）方案[139]。部署前，每个节点随机与最靠近自己期望位置的 c 个节点建立配对密钥。例如，对于节点 u 的邻居节点 v，部署服务器随机生成配对密钥 $k_{u,v}$，然后把$(v,k_{u,v})$和$(u,k_{u,v})$分别分配给 u 和 v。部署后，相邻节点通过交换节点标识符确定双方是否存在配对密钥。

CPKS 方案的优点是，每个节点仅与有限个相邻节点建立配对密钥，网络规模不受限制；配对密钥与位置信息绑定，任何节点的受损不会影响其他节点的安全。缺点是密钥连通概率的提高仅能通过分配更多的配对密钥来实现，受到一定的限制。

针对上述问题，Liu 提出了使用基于地理信息的对称二元多项式随机密钥预分配[139]（location-based key predistribution，LBKP）方案。该方案把部署目标区域划分为若干个大小一致的正方形区域。部署前，部署服务器生成与区域数量相等的对称 t 阶二元多项式，并

为每一区域指定唯一的二元多项式。对于每一节点,根据其期望位置来确定其所处区域,部署服务器把与该区域相邻的上、下、左、右 4 个区域以及节点所在的区域共 5 个二元多项式共享载入该节点。部署后,两个节点若共享至少 1 个二元多项式共享就可以直接建立配对密钥。该方案通过调整区域的大小来解决 CPKS 方案存在的连通概率受限的问题。与 E-G 方案和 q-composite 方案甚至 Blundo 方案相比,LBKP 方案的抗毁性明显提高,但缺点是计算和通信开销过大。

在基于部署知识的随机密钥预分配方案[140]中,假定网络的部署目标区域是一个二维矩形区域且节点部署服从 Gaussian 分布。节点被划分为 $t\times n$ 个部署组,每个组 $G_{i,j}$($i=1,2,\cdots,t$;$j=1,2,\cdots,n$)的部署位置组成一个栅格。密钥池(密钥数为$|S|$)被划分成若干个子密钥池(密钥数为$|S_c|$),每个子密钥池对应于一个部署组。若两个子密钥池是水平或垂直相邻,则至少共享 $a|S_c|$个密钥;若两个子密钥池是对角相邻,则至少共享 $b|S_c|$密钥(a,b 满足以下关系:$0<a,b<0,25$ 且 $4a+4b=1$)。若两个子密钥池不相邻,则没有共享密钥。如图 4.7.7 所示。

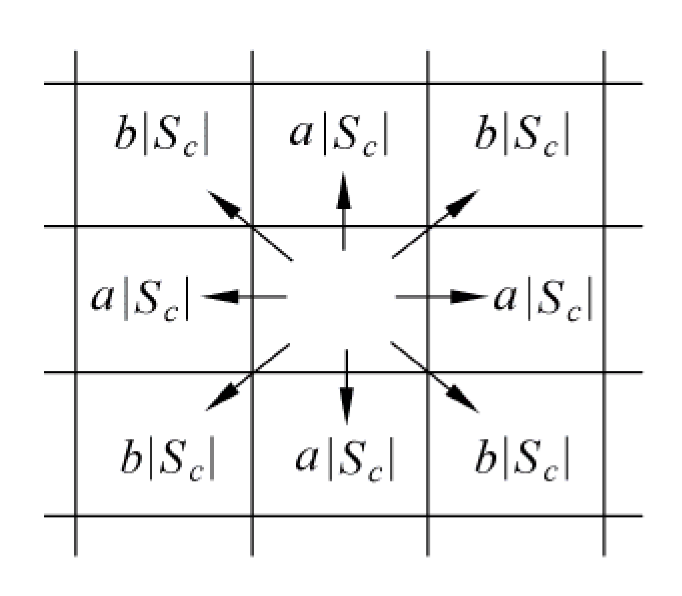


图 4.7.7 相邻密钥池之间的共享密钥数

对于组内每一节点,从对应的子密钥池随机取 m 个不同的密钥。部署后,若相邻节点存在共享密钥,则可以直接建立配对密钥。实验表明,在同等条件下,该方案提高了节点的连通概率。例如,当节点预分配的密钥数为 100 时,E-G方案的节点连通概率仅为 0.095,而该方案能够达到 0.687。使用部署知识使得节点减少了预分配无用密钥的数量,提高了网络抗毁性。但该方案的子密钥池的划分需要慎重考虑。

尽管 Liu[139]和 Du[140]都在密钥预分配时使用节点的位置信息以提高抗毁性,但存在着攻击者容易对节点进行定位后俘获以及节点因缺乏认证机制而被伪造等问题。针对上述问题,Huang 的栅格组部署方案[141]使用限制组的节点数量、设定密钥空间被选中的阈值等方法提出了解决方案。

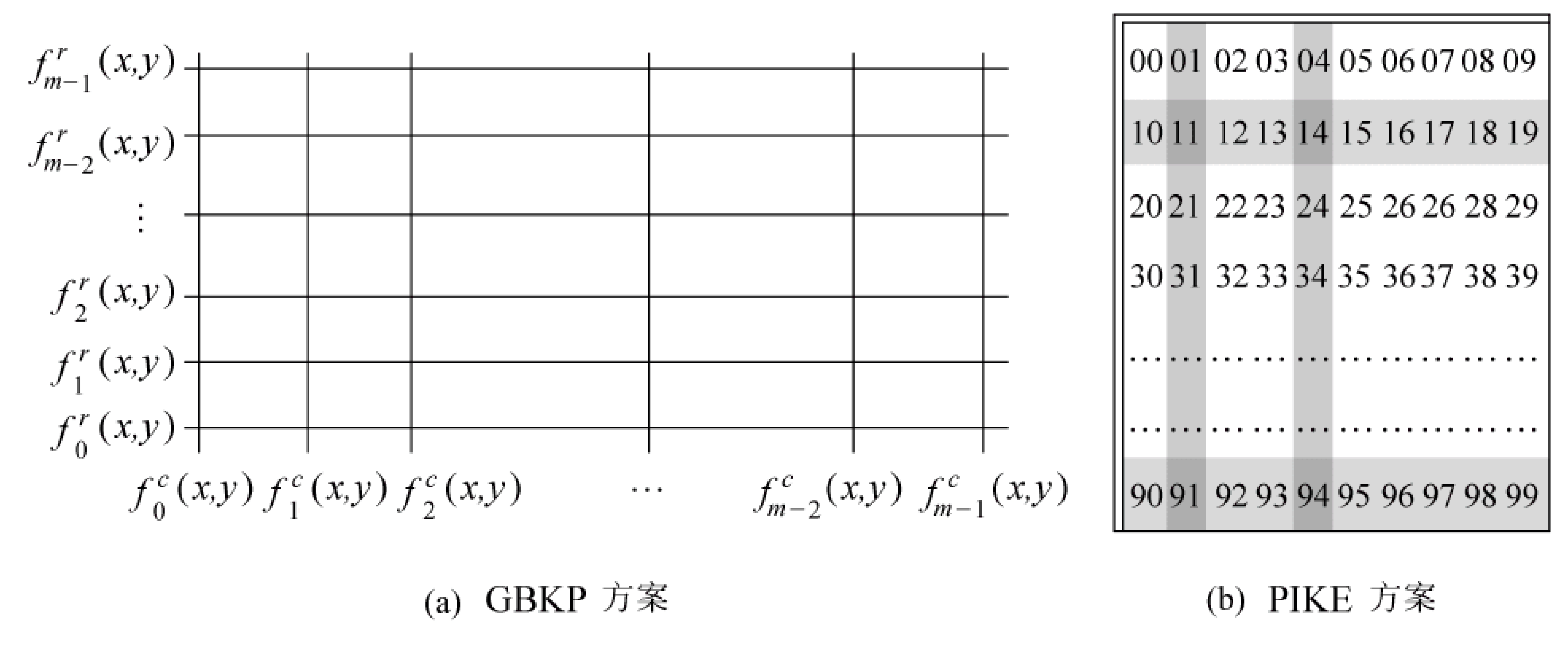
5. 多路径密钥增强方案[136]

在 E-G 方案里,两个相邻节点 A 和 B 所被分配的密钥有可能被分配给其他节点,若这些节点受损,则 A 和 B 之间的链路会受到安全威胁。Chan 提出了多路径密钥增强方案。假设 A 和 B 经过密钥协商后存在着 j 条不相交的路径,A 产生 j 个随机值 $v_1,v_2,\cdots,v_j$,然后通过 j 条不相交的路径发送给 B,B 接收到这 j 个随机值后,生成新的配对密钥 $K=k\oplus v_1\oplus v_2\oplus\cdots\oplus v_j$。攻击者若不能获取全部的 j 个随机值,则不能破译配对密钥 K。该方案若与 E-G 方案或其他随机密钥管理方案结合使用,则能够显著提高相应方案的安全性能。但该方案的缺点是,如何建立和能否建立足够数量的不相交路径在目前尚属于 NP 问题。

4.7.5.3 基于栅格的密钥预分配方案[138,142]

建立栅格的方法如下:根据网络中的节点总数 N 构造 $m\times m$ 个栅格,其中,$m=\lfloor\sqrt{N}\rfloor$。在 Liu 提出的 GBKP(grid-based key predistribution)方案[138]里,部署前,部署服务器生成 $2m$ 个多项式,栅格的每一行对应于唯一的一个多项式,每一列对应于另一个唯一的多项

式。部署服务器把节点逐一对应于各栅格的汇合点,并把对应的多项式共享和标识符配置给该节点,如图 4.7.8(a)所示;部署后,同一行或列的节点可以直接建立配对密钥,不同行列的节点通过中间节点建立密钥路径。而在 Chan 提出的 PIKE(peer intermediaries for key establishment)方案[142]里,节点按照栅格的行列号编号,部署前,每一节点都与同一行列共 $2(\sqrt{N}-1)$个其他节点建立配对密钥,然后节点按照序列号顺序进行部署,如图 4.7.8(b)所示;部署后,同一行或列的节点直接拥有配对密钥,不同行列的节点则通过公共行列的节点建立密钥路径。



(a) GBKP 方案　　(b) PIKE 方案

图 4.7.8　基于栅格的密钥预分配

GBKP 方案和 PIKE 方案都保证任意两个节点能够建立配对密钥,与节点密度无关,且能够显著降低节点的通信和存储开销。但其缺点是部署方式固定,不够灵活,中间节点的受损会影响整个网络的安全。

4.7.5.4 基于组合论的密钥预分配方案[143]

Camtepe 把组合设计理论(combinatorial design theory)用于设计 WSN 确定密钥预分配方案上。假设网络的节点总数为 N,用 n 阶有限射影空间(finite projective plane)(n 为满足 $n^2+n+1\geqslant N$ 的素数)生成一个参数为($n^2+n+1, n+1, 1$)的对称 BIBD,支持的网络节点数为 n^2+n+1,密钥池的大小为 n^2+n+1,能够生成 n^2+n+1 个大小为 $n+1$ 的密钥环,任意两个密钥环至少存在 1 个公共密钥,并且每一密钥出现在 $n+1$ 个密钥环里。可见,任意两个节点的密钥连通概率为 1。但素数 n 不能支持任意的网络规模。例如,当 $N>n^2+n+1$ 时,n 必须是下一个新的素数,而过大的素数则会导致密钥环急剧增大,突破节点的存储空间而不适用于 WSN。使用广义四角形(generalized quadrangles,GQ)可以更好地支持网络规模,如 GQ(n,n),GQ(n,n^2)和 GQ(n^2,n^3)分别支持的网络规模达 $O(n^3)$,$O(n^5)$和 $O(n^8)$,但也存在着素数 n 不容易生成的问题。

为此,Camtepe 提出了对称 BIBD 与 GQ 相结合的混合密钥预分配方案:使用对称 BIBD 或 GQ 生成 b 个(b 值大小由 BIBD 或 GQ 决定,$b<N$)密钥环,然后使用对称 BIBD 或 GQ 的补集设计(complementary design)随机生成 $N-b$ 个密钥环,与前面生成的 b 个密钥环一起组成 N 个密钥环。这种混合的密钥预分配方案提高了网络可扩展性和抗毁性,但不保证节点的密钥连通概率为 1。无论是对称 BIBD、GQ 还是混合方案,都有比 E-G 方案更

高的密钥连通概率,平均密钥路径长度也更短。

4.7.5.5 SPINS 协议[144]和 LEAP 协议[145]

Perrig 利用 sink 作为网络的可信密钥分发中心为网络节点建立配对密钥及实现对广播数据包的认证。SPINS(security protocols for sensor networks)协议由两部分组成:SNEP(secure network encryption protocol)和 μTESLA(timed efficient stream loss-tolerant authentication)。SNEP 主要通过使用计数器(counter)、消息认证码 MAC (message authentication code)等机制来实现数据的机密性及数据认证。通信双方的配对密钥及 MAC 密钥都通过使用从 sink 获取的主密钥及伪随机函数生成。SNEP 使得协议达到语义级安全(相同的明文在不同的时段加密,其密文不相同),保证了数据的鲜活性;MAC 密钥长度固定,仅为 8 字节,不增加过多的通信负载。

μTESLA 实现对广播数据的认证。sink 首先使用单向散列函数 H 生成一个单向密钥链$\{K_0, K_1, \cdots, K_n\}$,其中,$K_i = H(K_{i+1})$,由 K_{i+1}很容易计算得到 K_i,而由 K_i 则无法计算得到 K_{i+1}。网络运行时间分为若干个时间槽(slot),在每一个时间槽使用密钥链里对应的一个密钥。在第 i 个时间槽里,sink 发送认证数据包,然后延迟一个时间 δ 后公布密钥 K_i。节点接收到该数据包后首先保存在缓冲区里,并等待接收到最新公布的密钥 K_i,然后使用其目前保存的密钥 K_v,并使用 $K_i = H^{i-v}(K_v)$来验证密钥 K_i 是否合法,若合法,则使用 K_i 认证缓冲区里的数据包。

μTESLA 工作示意图如图 4.7.9 所示。在 μTESLA 里,攻击者很难获取或伪造最新的认证密钥。因此,μTESLA 提供了良好的广播认证机制。但密钥延迟暴露和非实时认证的问题,使其很容易受到 DoS 攻击。针对这些问题,Liu 分别提出了使用多级 μTESLA[152] 和 Merkle 散列树[153]的解决方法。

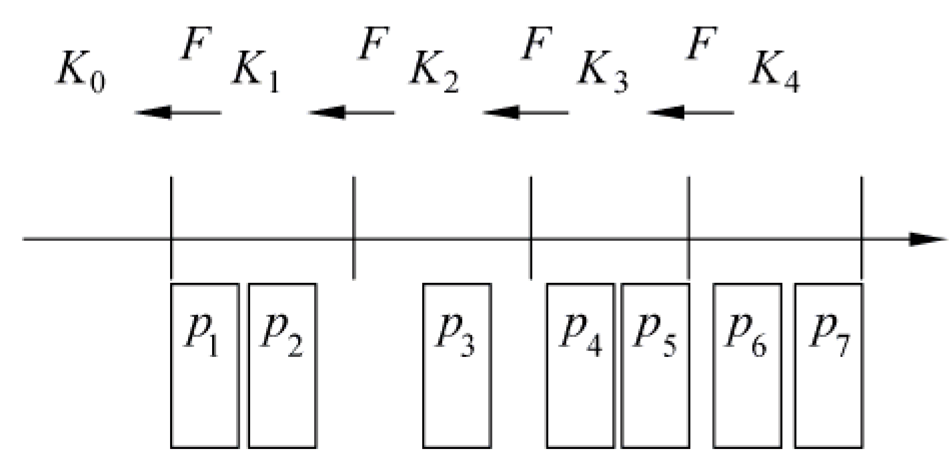


图 4.7.9 μTESLA 单向密钥链[144]

在 SPINS 协议里,任何节点的配对密钥生成、数据包认证都必须通过 sink 来完成。一旦 sink 受损,则整个网络的安全都受到威胁。而且 sink 开销过大,SPINS 协议仅适用于规模较小的网络。

Zhu 认为,任何一种单一的密钥机制都不可能实现 WSN 所需的安全通信,因此提出 LEAP(localized encryption and authentication protocol)协议[145],建立了 4 种类型的密钥:个体密钥、配对密钥、组密钥和簇密钥。个体密钥为节点与 sink 共享的密钥,由节点在部署前通过预分配的主密钥和伪随机函数来生成。若两个相邻节点要生成配对密钥,则通过交换其标识符及使用预分配的主密钥和单向散列函数计算得到。若节点作为簇头要建立与其邻居节点共享的簇密钥,则产生一个随机密钥作为簇密钥,然后使用与邻居节点的配对密钥逐一地对簇密钥加密后发送给对应节点,邻居节点把簇密钥解密后保存下来。组密钥为 sink 与所有节点共享的通信密钥。sink 首先把组密钥使用与其子节点共享的簇密钥加密后广播给子节点,子节点获取最新的组密钥后,用与其下一级子节点共享的簇密钥加密组密钥后广播给其子节点。依此类推,直到所有节点都获取最新的组密钥为止。

LEAP 协议的优点是任何节点的受损都不会影响其他节点的安全,缺点是节点部署后,

在一个特定的时间内必须保留全网通用的主密钥。若主密钥一旦被暴露，则整个网络的安全都受到威胁。

4.7.5.6　基于 EBS 的动态密钥管理方案[146]

EBS(exclusion basis systems)由 Eltoweissy 提出，主要用于密钥动态管理[154]。EBS 为一个三元组(n,k,m)表示的集合Γ，其中，n 为组的用户数，k 为节点存储的密钥数，m 为密钥更新的信息数。对于任一整数(用户)$t\in[1,n]$，具有以下属性：① t 最多出现在Γ 的 k 个子集(密钥)里，表示任一用户最多拥有 k 个密钥；②有 m 个子集(密钥)，$A_1,A_2,\cdots,A_m$，满足 $\bigcup_{i=1}^{m} A_i=[1,n]-\{t\}$，表示使用 m 个与 t 无关的密钥更新信息可撤回用户 t。

Younis 在层次式 WSN 里提出基于位置信息的 EBS 动态密钥管理方案 SHELL (scalable, hierarchical, efficient, location-aware, and light-weight)[146]。在 SHELL 方案里，普通节点按照其地理位置被划分为若干簇，由簇头，或称为网关(gateway)节点来控制。网关节点有可能被命令节点指定为其他簇的密钥生成网关节点(key generating gateway)。它并不存储和生成自己簇里各节点的管理密钥。根据簇数和节点的存储容量，簇 C_i 的网关节点 $G_{\mathrm{CH}}[i]$使用正则矩阵法生成所在簇的(n,k,m)——EBS 矩阵，并把矩阵的相关部分内容分别发送给该簇的密钥生成网关节点 $G_{K_1}[i]$和 $G_{K_2}[i]$等。密钥生成网关节点根据 EBS 矩阵的内容生成相应的管理密钥，并通过网关节点 $G_{\mathrm{CH}}[i]$广播给簇内各节点。为了避免串谋攻击，相邻节点管理密钥的汉明距(Hamming distance)设计为最小。

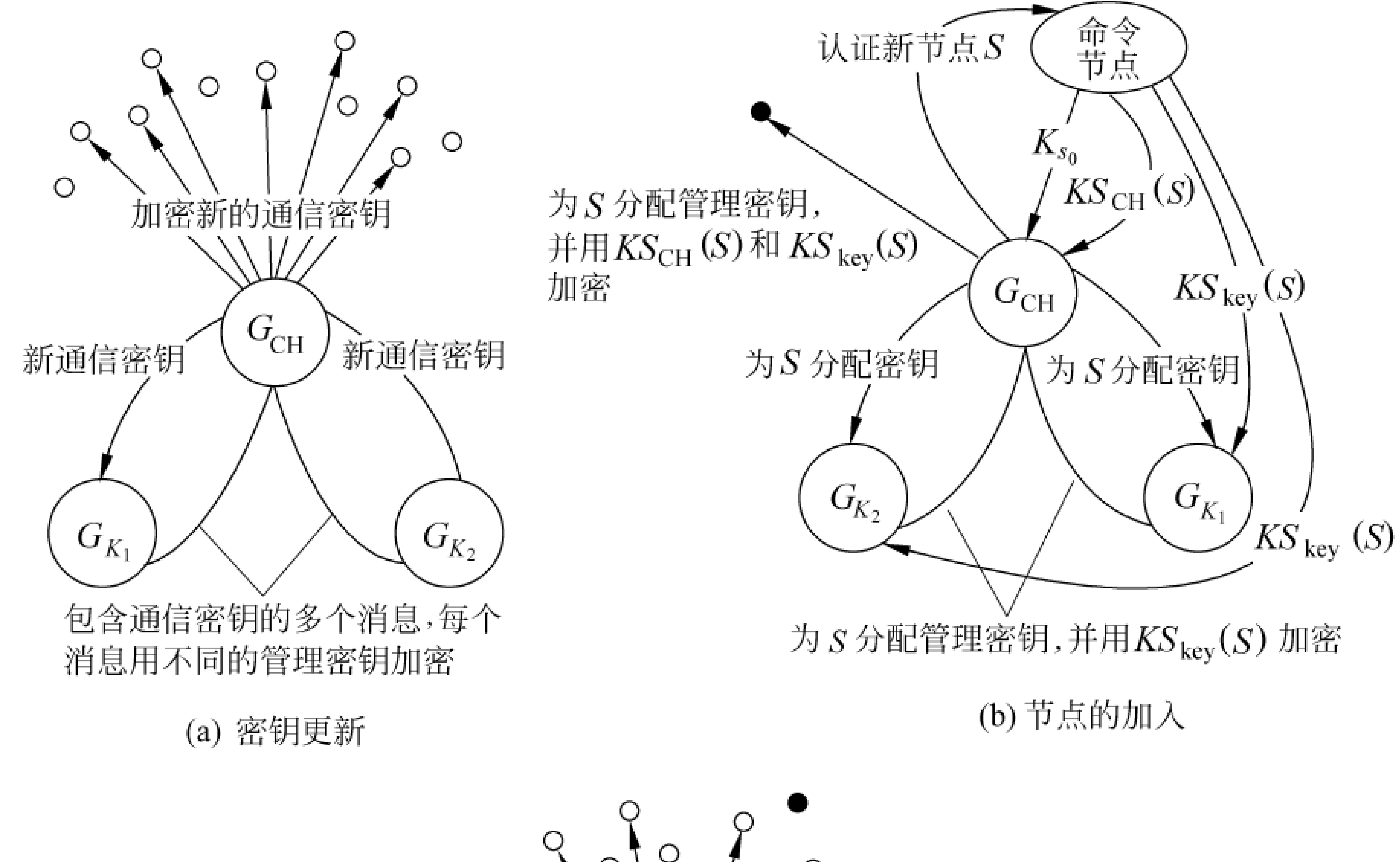
SHELL 定期更新密钥。当需要更新密钥时，由簇头首先把最新的通信密钥发送给密钥网关生成节点，然后由密钥网关生成节点生成新的管理密钥，再通过簇头发送给簇内各节点，如图 4.7.10(a)所示。

当新的节点加入时，首先根据其地理位置确认加入所在簇，并通过命令节点认证其身份，然后由簇头与密钥生成网关节点协调启动管理密钥生成进程，如图 4.7.10(b)所示。当要撤回受损的节点时，若是簇头受损，则可以采取指定新的簇头或把簇内节点重新分配到其他正常的簇内等方法；若是普通节点受损，簇头把受损节点信息通知密钥生成网关节点，然后由密钥网关生成节点利用 EBS 的性质生成新的管理密钥，并通过簇头广播发送给簇内节点，受损节点由于无法解密广播数据包而无法获取新的管理密钥，如图 4.7.10(c)所示。

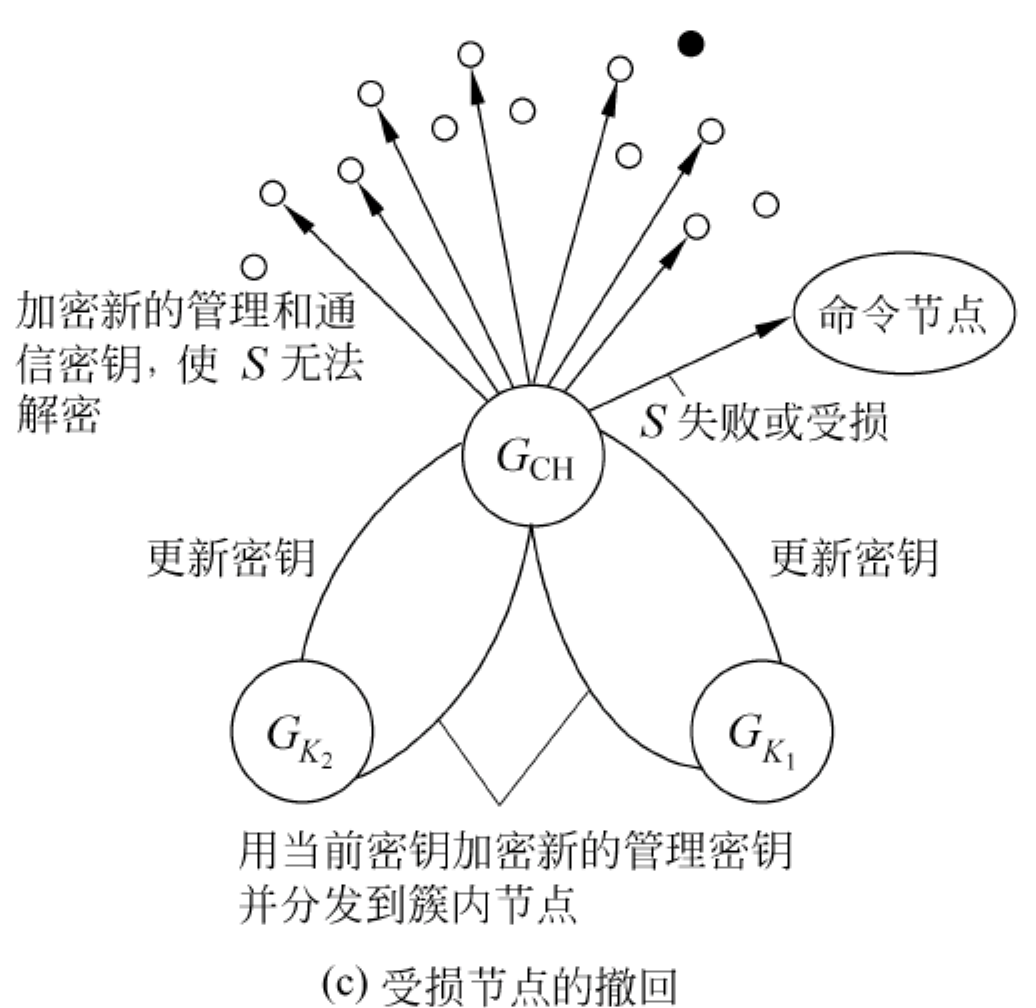
与随机密钥分配方案相比，SHELL 明显增强了抗串谋攻击的能力。例如，当 $k=4$，$n=200$ 时，若要发起串谋攻击，则在 SHELL 里需要使 11 个节点受损，而在随机密钥分配方案时仅需 3 个节点受损。但在 SHELL 里由密钥网关生成节点存储相应簇的节点密钥，这意味着，密钥网关生成节点受损数量越多，网络机密信息暴露的可能性就越大。针对 SHELL 的缺点，Eltoweissy 提出了 LOCK(localized combinatorial keying)方案[147]。该方案使用两层 EBS 管理密钥对基站、簇头和普通节点的密钥分配、更新、撤回进行管理，使得簇头的受损不会暴露更多机密信息。

4.7.5.7　对称与非对称混合密钥管理协议

在基于证书密码体制 CBC (certificate-based cryptography) 的 PKC (public key cryptography)涉及的一个基本问题是公钥的认证，即在使用对方节点的公钥加密时，必须



(a) 密钥更新　　(b) 节点的加入



(c) 受损节点的撤回

图 4.7.10　密钥更新、节点的加入与受损节点的撤回

先对公钥进行认证。Huang 提出使用椭圆曲线密码体制 ECC(elliptic curve cryptography)与对称密钥的混合密钥管理协议[155]来解决异构节点之间的公钥认证问题。

在异构 WSN 里，FFD(full-functional devices)被认为具有较强的计算和通信能力，而 RFD(reduced-functional devices)的能力则比较受限。部署前，首先通过有限域 GF(q)上的一条椭圆曲线及相关信息生成隐式证书(implicit certificate)和 FFD 节点、RFD 节点各自的公/私密钥。部署后，FFD 节点和 RFD 节点通过对方的隐式证书获取相应的公钥，然后各自随机生成链路密钥的基值，并使用对方公钥加密后发送给对方。若双方的链路密钥基值都得到验证，则与标识符 ID 一起共同协商生成链路密钥，FFD 节点与 RFD 节点就使用生成的链路密钥进行安全通信。在该方案中，FFD 节点和 RFD 节点都提供链路密钥的基值，但最终链路密钥是通过密钥派生函数生成的，因此，FFD 节点和 RFD 节点都不能完全控制对链路密钥的选择，攻击者为了获取私钥所付出的代价比解决椭圆曲线的离散对数问题(discrete logarithm problem，DLP)所付出的代价还要大。该协议提供了隐式和显式的密钥

验证，这样就能确保只要在运行期间不出错，双方接收到的信息都是正确的。

该协议把ECC所产生的计算开销大都集中在FFD节点，未过多增加RFD节点的计算和通信开销。Kotzanikolaou对该协议进行了功能扩展[156]，允许任意两个同构节点之间建立链路密钥。

4.7.5.8 IBC密钥预分配方案[157]

与CBC相比，基于身份密码体制(identity-based cryptography，IBC)[158]的主要优点是节点的公钥由公开信息直接推导获得，无需对公钥进行认证，从而有效地降低了计算复杂度和通信负载，被认为比较适用于WSN。Zhang提出了将Bilinear Pairing技术与地理信息相结合的IBC密钥管理方案。

部署前，节点A预加载系统参数($p,q,E/F_q,G_1,G_2,\hat{e},H,h,W,W_{pub}$)以及私钥$IK_A$，其中，$p$和$q$为有限域$F_q$的两个素数，$E$为$F_q$上的椭圆曲线，$G_1$和$G_2$分别是$F_q$上的加法群和乘法群，$\hat{e}$为双线性映射，$W$是在$G_1$上随机选取的生成元，$H$和$h$为两个散列函数，$W_{pub}=\kappa W$($\kappa$是主密钥)，$IK_A=\kappa H(ID_A)$。部署后，节点$A$通过定位算法获取其位置信息$l_A$后计算其私钥$LK_A=\kappa H(ID_A \parallel l_A)$。节点$A$与邻居节点$B$可以通过获取公开的ID和对方的地理信息来建立配对密钥$K_{A,B}$：A生成配对密钥$K_{A,B}=\hat{e}(LK_A,H(\mathrm{ID}_B \parallel l_B))$，$B$生成配对密钥$K_{B,A}=\hat{e}(LK_B,H(ID_A \parallel l_A))$，根据映射$\hat{e}$的性质可知，$K_{A,B}=\hat{e}(\kappa H(ID_A \parallel l_A),H(ID_B \parallel l_B))=\hat{e}(H(ID_A \parallel l_A),\kappa H(ID_B \parallel l_B))=K_{B,A}$，从而建立双方的配对密钥，在此基础上，使用$h$和配对密钥就可以生成通信所需的各种类型的会话密钥。

该方案具有很强的容侵能力，任何节点的受损都不会暴露其他节点的机密信息。由于采用可靠的节点间认证机制，从而有效防止了Wormhole、Sinkhole、Sybil和Bogus Data Injection等攻击。但该方案也存在一些缺陷：一是在节点预分配的主密钥κ必须等待私钥生成后才能删除，若主密钥被暴露，则整个网络的机密信息都会暴露；二是因其位置是固定的，因而仅适用于静态WSN；三是对节点资源的使用需求高，制约了其应用范围。

4.7.6 方案和协议的综合分析与所需解决的研究问题

从研究现状看，随机密钥预分配方案或协议被认为是最适用于WSN的，目前是WSN密钥管理的一个主流研究方向。表4.7.1在密钥池结构、密钥连接概率、抗毁性等方面对部分典型的随机密钥管理方案进行了比较。较高的密钥连通概率意味着相邻节点甚至全网络都可以达到较高的安全连通性；而密钥被暴露的概率越小，则意味着抗毁性就越好。在表4.7.1中，“↓”表示下降，“↑”表示上升，而“—”表示不变。

将密钥池设计为结构化、提高共享密钥阈值、利用地理信息或部署知识，可以有效地提高随机密钥预分配方案或协议的抗毁性。密钥连通概率与节点部署密度相关，其理论基础为经典的随机图模型。但文献[159]指出，经典随机图模型并不完全适用于WSN，同时也指出如何选取适当的密钥池及密钥环大小，以确保获取较高的密钥连接概率。文献[160]也针对随机图模型以及若干个随机密钥预分配方案进行了深入分析。

表4.7.2通过处理复杂度、通信复杂度、存储复杂度以及网络可扩展性等性能指标比较了一些典型的密钥管理方案和协议。

表 4.7.1　随机密钥管理方案和协议的比较

方案和协议	密钥池结构	密钥连通性	抗　毁　性	与 E-G 方案对比	
				密钥连通性	抗毁性
E-G scheme	Non-Structure	$1-\frac{((P-k)!)^2}{(P-2k)!\ P!}$	$1-\left(1-\frac{k}{P}\right)^x$	—	—
q-Composite scheme	Non-Structure	$1-\sum_{i=0}^{q-1}\frac{\binom{\lvert S\rvert}{i}\binom{\lvert S\rvert-i}{2(m-i)}\binom{2(m-i)}{m-i}}{\binom{\lvert S\rvert}{m}^2}$	$\sum_{i=q}^{m}\left(1-\left(1-\frac{m}{\lvert S\rvert}\right)^x\right)^i\frac{p(i)}{p}$	↓	↑
Multiple-Space key pre-distribution scheme	Structured	$1-\frac{((\omega-\tau)!)^2}{(\omega-2\tau)!\omega!}$	$\leqslant\omega\sum_{j=\lambda+1}^{x}\binom{x}{j}\left(\frac{\tau}{\omega}\right)^j\left(1-\frac{\tau}{\omega}\right)^{x-j}$	↓	↑
Polynomial-Based key predistribution scheme	Structured	$1-\prod_{i=0}^{s'-1}\frac{s-s'-i}{s-i}$	$1-\sum_{i=0}^{t}\binom{N_c}{i}\left(\frac{s'}{s}\right)^i\left(1-\frac{s'}{s}\right)^{N_c-i}$	↑	↑
CPKS	Structured	$\frac{c}{m}\iint_{(x-i_x)^2+(y-i_y)^2\leqslant d'^2}\frac{p(v_{j_x,j_y},u_{i_x,i_y})}{\pi d'^2}\mathrm{d}x\mathrm{d}y$	0	↑	↑
LBKP	Structured	$\frac{\sum_{C_{j_c,j_r}\in S_{i_c,i_r}}p(C_{j_c,j_r},C_{i_c,i_r})}{\sum_{\forall j_c,j_r}p(C_{j_c,j_r},C_{i_c,i_r})}$	$1-\sum_{i=1}^{t}\binom{N_s}{i}p_c^i(1-p_c)^{N_s-i}$	↑	↑
Key pre-distribution scheme using deployment knowledge	Structured	$1-\frac{\sum_{i=0}^{\min(m,\lambda\lvert S_c\rvert)}\binom{\lambda\lvert S_c\rvert}{i}\binom{(1-\lambda)\lvert S_c\rvert}{m-i}\binom{\lvert S_c\rvert-i}{m}}{\binom{\lvert S_c\rvert}{m}^2}$	$1-\left(1-\frac{m}{\lvert S\rvert}\right)^x$	↑	—
Grid-Group deployment scheme	Structured	$1-\frac{((\omega-\tau)!)^2}{(\omega-2\tau)!\omega!}$	λ-secure	↓	↑

表 4.7.2　密钥管理方案和协议的性能比较

方案与协议	计算复杂性	通信复杂性	存储复杂性	扩展性
E-G scheme	$O(k)$	$O(2)$	$O(k)$	Good
q-Composite scheme	$O(m)$	$O(2)$	$O(m)$	Moderate
Multiple-Space key pre-distribution scheme	$O(\lambda)$	$O(2)$	$O(\lambda\tau)$	Weak
Polynomial-Based key predistribution scheme	t modular multiplication and t modula addition	$O(2)$	$O(t\log q)$	Weak
CPKS	0	0	$O(c)$	Strong
LBKP	t modular multiplication and t modular addition	$O(2)$	$O(t\log\text{q})$	Moderate
Key pre-distribution scheme using deployment knowledge	$O(m)$	$O(2)$	$O(m)$	Moderate
Grid-Group deployment scheme	$O(\lambda)$	$O(\tau)$	$O(\lambda\tau)$	Moderate
Grid-Based key predistribution	t modular multiplication and t modular addition	$O(2)$	$O(t\log q)$	Weak
PIKE	$O(2)$	$O(\sqrt{n})$	$O(\sqrt{n})$	Weak
Hybrid designs for scalable key distributions	$O(n)$	$O(d)$	$O(n)$	Moderate
SNEP	1 encryption and 1 MAC computation	$O(2)$	8bytes	Weak
μTESLA	1 hash computation	0	High	Weak
LEAP	$O\left(\frac{d^2}{N}\right)$	$O(\log N)$	$O(d+L)$	Weak
SHELL	High	High	$O(k)$	Moderate
LOCK	High	High	$O(k)$	Moderate
Fast authenticated key establishment protocols	760ms	1437bytes	5.2kbytes	Moderate
Location-Based key management scheme	62.04ms	84bytes	High	Strong

随机密钥预分配方案或协议虽然不能提供最佳的密钥连通概率，但其计算、存储和通信开销较为理想，且具有良好的分布特性。而确定密钥预分配或非对称密钥管理方案和协议虽然可以保证任何两个节点都能建立密钥连接，但计算、存储和通信开销大的问题仍需进一步优化。

总之，虽然密钥管理的研究取得了许多成果，但密钥管理的方案和协议仍然不能满足各种应用需求，还存在一些需要解决的问题。具体如下：

（1）建立多种类型的通信密钥。目前的 WSN 密钥管理方案和协议大多仅考虑建立邻居节点间的配对密钥，但配对密钥只能实现节点一对一通信，不支持组播或全网广播。方案或协议应建立多种类型通信密钥，满足单播通信、组播通信或广播通信等需求。

(2) 支持密钥的分布式动态管理。节点的受损是不可避免的,若要把受损节点排除于网络之外,首先要动态更新或撤回已受损的密钥,但目前的大多数方案或协议较少考虑密钥动态管理问题。已有的密钥动态管理方案多以集中式为主,产生了过多的计算和通信开销。密钥更新和撤回应以节点之间的协作实现为主,才能使方案或协议具有良好的分布特性[161]。

(3) 提供有效的认证机制。密钥的协商需要对数据包和节点身份进行有效认证,否则不能保证所建立的通信密钥的正确性。单纯的 MAC 机制在对称密钥管理中存在被伪造的问题,基于非对称密钥的数字签名机制目前还不适用于 WSN。提供符合 WSN 特点的认证机制是密钥管理研究的重要内容。

(4) 支持容侵和容错。节点易受损及计算通信能力受限的特点,使得节点很容易受到 DoS 攻击[162],全面防御 DoS 攻击是比较困难的。此外,即使未受到安全威胁,节点出于对节能的考虑或因资源被耗尽导致不能保证永远处于正常运行状态,数据包的丢失在所难免。因此,方案和协议应具有良好的容侵和容错性。

从体系结构的观点来看,密钥管理要为其他安全机制提供基础服务,并与这些安全机制共同组成 WSN 的整体安全解决方案。我们认为,实现跨层设计的密钥管理将有利于明确设计目标及性能优化。例如,目前绝大多数的密钥管理方案和协议都仅仅致力于建立相邻节点之间的通信密钥,而在一些有效的安全解决方案[163]里,多跳节点之间的通信密钥也是必要的。加强密钥管理与安全路由、安全定位、安全数据融合等安全机制的耦合,就能够从系统整体的角度对方案和协议的处理复杂度、存储复杂度和通信复杂度进行优化,从而使得所设计的密钥管理方案和协议更加符合 WSN 特点,具有良好的适应性。

运用符合 WSN 特点的理论分析方法进行密钥管理的研究是十分必要的,这样能够避免所设计的机制和算法过多地依赖直觉经验而缺乏严谨的、科学的、可信的理论依据,从而避免研究成果的片面性、局部化,甚至不可用。为了提供更加有效的解决方案,我们将依靠成熟且可行的理论方法,如随机图理论、信息论等理论方法,采用 WatchDog[164]、单向散列函数/链、self-healing 技术[165]等安全算法和技术,结合 WSN 的资源受限、拓扑易变、部署随机、自组织、规模大、无固定设施支持等特点,设计可行、可靠的密钥管理方案或协议,实现密钥管理机制和算法的可模型化、可度量化和可计算。

参 考 文 献

1. Akyildiz I F, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor network: A survey. Computer Networks, 2000, 38: 393～422
2. Canetti R, Malkin T, Nissim K. Efficient communication-storage tradeoffs for multicast encryption. In: Stem J, ed. Proceedings of Advances in Cryptology-EUROCRYPT'99, LNCS 1599, New York: Springer-Verlag, 1999. 59～474
3. Perrig A, Song D, Tygar J D. ELK: A new protocol for efficient large-group key distribution. In: Proceedings of the IEEE Symposium on Security and Privacy, 2001. 247～262
4. Li M Y, Poovendran R, Bernstain C. Optimization of key storage for secure multicast. Conference on Information Science and Systems 2001, 2001. 771～774
5. Fiat A, Naor M. Broadcast encryption. In: Proceedings of Adances in Crypto'92, LNCS 839,

Springer-Verlag，1994. 257～270

6. Eskicioglu M. Multimedia security in group communications: Recent progress in wired and wireless networks. In: Proceedings of IASTED Int'l Conference on Communications and Computer Networks, 2002. 125～133
7. Rafaeli S, Hutchison D. A survey of key management for secure group communication. ACM Computing Surveys, 2003, 35(3): 309～329
8. NIST. NIST: FIPS 186 for digital signature standard (DSS), May 1994
9. Wong C, Lam S. Keystone: A group key management service. In: Proceedings of 1st International Conference on Telecommunications (ICT), May 2000
10. Staddon J, Miner S, Franklin M, Balfanz D, Malkin M, Dean D. Self-healing key distribution with revocation. In: Proceedings of IEEE Symposium on Security and Privacy, 2002. 241～257
11. Naor D, Naor M, Lotspiech J. Revocation and tracing schemes for stateless receivers. In: Proceedings of Advances in Cryptology (CRYPTO 2001), LNCS 2139, Springer-Verlag, 2001. 41～62
12. Naor M, Pinkas B. Efficient trace and revoke schemes. In: Proceedings of 4th International Conference on Financial Cryptography, 2000. 1～20
13. Zhu S, Setia S, Jajodia S. Adding reliable and self-healing key distribution to the subset difference group rekeying method for secure multicast. George Mason University Technical Report ISETR-03-02, April 2003
14. Trappe W, Song J, et al. Key management and distribution for secure multimedia multicast. IEEE Trans on Multimedia, 2003, 5(4): 544～557
15. Wallner D M, Harder E J, Agee R C. Key Management for Multicast: Issues and Architectures. RFC 2627, June 1999
16. Mittra S. Iolus: A framework for scalable secure multicasting. In: Proceedings of the ACM SIGCOMM, 1997. 277～288
17. Wong C K, Gouda M G, Lam S S. Secure group communications using key graphs. IEEE/ACM Trans on Networking, 2000, 8(1): 16～30
18. Mcgrew D A, Sherman A T. Key establishment in large dynamic groups using one-way function trees. Technology Report No. 0755. TIS Labs at Network Associates, Inc.
19. Waldvogel M, Caronni G, Sun D, Weiler N, Plattner B. The VersaKey framework: Versatile group key management. IEEE Journal on Selected Areas in Communications(Special Issue on Middleware), 1999, 17(9): 1614～1631
20. Chang I, Engel R, Kandlur D, Pendarkis D, Saha D. Key management for secure Internet multicast using Boolean function minimization techniques. In: Proceedings of IEEE INFOCOM, 1999, 2: 689～698
21. Harney H, Muckenhirn C. Group Key Management Protocol (GKMP) specification. RFC 2093. 1997
22. Harney H, Muckenhirn C. Group Key Management Protocol (GKMP) architecture. RFC 2094. 1997
23. Canetti R, Garay J, Itkis G, Micciancio D, Naor M. Multicast Security: A taxonomy and some efficient constructions. In: Proceedings of IEEE INFOCOM, 1999. 708～716
24. 李彦希，林闯，尹浩，蒋屹新. 基于单向函数树的高效分布式组密钥管理方案. 清华大学学报. 2005, 45(10): 1417～1420
25. Blum M, Micali S. How to generate cryptographically strong sequences of pseudo-random bits. SIAM J on Computer, 1984, 13: 850～864
26. Chan K C, Chan S H. Key management approaches to offer data confidentiality for secure multicast.

IEEE Network，2003，11：30～39

27. Ballardie A. Scalable multicast key distribution. RFC 1949，1996
28. Decleene B，Dondeti L，Griffin S，Hardjono T，Kiwior D，Kurose J，Towsley D，Vasudevan S，Zhang C. Secure group communications for wireless networks. In：Proceedings of IEEE MILCOM，June 2001
29. Dondeti L，Mukherjee S，Samal A. Scalable secure one-to-many group communication using dual encryption. Computer Communications，1999，23(17)：1681～1701
30. Rafaeli S，Hutchison D. Hydra：A decentralized group key management. In：Proceedings of the 11th IEEE International WETICE：Enterprise Security Workshop，Los Alamitos，CA：IEEE Computer Society Press，2002
31. Setia S，Koussih S，Jajodia S，Harder E. Kronos：A scalable group re-keying approach for secure multicast. In：Proceedings of the IEEE Symposium on Security and Privacy，2000. 215～228
32. Mittra S. Iolus：a framework for scalable secure multicasting. In：Proceedings of the ACM SIGCOMM，1997. 277～288
33. Molva R，Pannetrat A. Scalable multicast security in dynamic groups. In：Proceedings of the 6th ACM CCS，1999. 101～112
34. Dondeti L R，Mukherjee S. DISEC：A distributed framework for scalable secure many-to-many communication. In：Proceedings of the 5th IEEE Symposium on Computers and Communications，IEEE Computer Society Press，2000. 693～698
35. Steiner M，Tsudik G，Waidner M. Diffie-Hellman key distribution extended to group communication. In：Proceedings of the 3rd ACM CCS，1996. 31～37
36. Steiner M，Tsudik G，Waidner M. Cliques：A new approach to group key agreement. In：Proceedings of the 18th International Conference on Distributed Computing Systems (ICDCS'98)，1998. 380～387
37. Diffie W，Hellman M E. New directions in cryptography. IEEE Trans on Information Theory，1976，22(6)：644～654
38. Kim Y，Perrig A，Tsudik G. Simple and fault-tolerant key agreement for dynamic collaborative groups. In：Proc ACM Conference on Computer and Communication Security，2000. 235～244
39. Perrig A. Efficient collaborative key management protocols for secure autonomous group communication. In：Proceedings of the International Workshop on Cryptographic Techniques and E-Commerce (CrypTEC'99). City University of Hong Kong Press，Hong Kong，China. 1999. 192～202
40. Becker C，Wille U. Communication complexity of group key distribution. In：Proceedings of the 5th ACM CCS，1998. 1～6
41. Boyd C. On key agreement and conference key agreement. In：Proceedings of the Information Security and Privacy：Australasian Conference. LNCS 1270，New York：Springer-Verlag，1997. 294～302
42. Burmester M，Desmedt Y. A secure and efficient conference key distribution system. In：Proceedings of Advances in Cryptology (Eurocrypt'94)，LNCS 950，New York：Springer-Verlag，1994. 275～286
43. Kim Y，Perrig A，Tsudik G. Communication-efficient group key agreement. In：Proceedings of IFIP SEC，2001
44. Project JXTA，http://www.jxta.org. Accessed：April 10. 2006
45. Seamons K E，Winslett M，Yu T. Limiting the disclosure of access control policies during automated trust negotiation. In：Proceedings of Network and Distributed Systems Security Symposium，2001
46. Winsborough W H，Li N. Towards practical automated trust negotiation. In：Proceedings of IEEE the 3rd International Workshop on Policies for Distributed Systems and Networks，2002. 92～103
47. Virgil D，Gligor H，Khurana R，et al. On the negotiation of access control policies. In：Proceedings of

IEEE the 2nd International Workshop on Policies for Distributed Systems and Networks, 2001. 188～201
48. Varshney U. Multicast over wireless networks. Communications of the ACM, 2002, 45(12): 31～37
49. Chang I, Engel R, Kandlur D, Pendarkis D, Saha D. Key Management for secure Internet multicast using Boolean function minimization techniques. In: proceedings of IEEE INFOCOM, 1999, 2: 689～698
50. Yang Y, Li X, Zhang X, et al. Reliable group rekeying: Design and performance analysis. ACM SIGCOMM 2001, San Diego, CA, USA. 2001. 27～38
51. Challal Y, Seba H. Group key management protocols: A novel taxonomy. International Journal of Information Technology, 2005, 2(1): 105～119
52. Briscoe B. MARKS: Multicast key management using arbitrarily revealed key sequences. In: Proceedings of the 1st International Workshop on Networked Group Communication, Pisa, Italy, Nov. 1999
53. Bohio M J, Miri A. Self-healing in group key distribution using subset difference method. In: Proceedings of the 3rd IEEE International Symposium on Network Computing and Applications, 2004. 405～408
54. Nakamura Y, Kikuchi H. Efficient key management based on the subset difference method for secure group communication. In: Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05), 2005. 707～712
55. Blundo C, D'Arco P, Listo M. A new self-healing key distribution scheme. In: Proceedings of IEEE Symposium on Computers and Communications (ISCC 2003), 2003. 803～808
56. Blundo C, D'Arco P, De Santis A, Listo M. Design of self-healing key distribution schemes. Design, Codes, and Cryptography, 2004, 32: 15～44
57. Wang P, Ning P, Reeves D S. Storage-efficient stateless group key revocation. In: Proceedings of the 7th International Conference on Information Security, ISC 2004. 25～38
58. Liu D, Ning P, Sun K. Efficient self-healing group key distribution with revocation capability. In: Proceedings of the 10th ACM CCS, 2003. 231～240
59. 朱雪龙. 应用信息论基础. 北京: 清华大学出版社, 2001
60. Liu Donggang, Ning Peng. Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks. In: Proceedings of the 10th Annual Network and Distributed System Security Symposium. San Diego, California. February 2003. 263～276
61. Ramkumar M, Memon N, Simha R. Pre-loaded key based multicast and broadcast authentication in mobile ad-hoc networks. In: Proceedings of IEEE Globecom 2003, San Francisco, CA, Dec 2003. 1405～1409
62. Ramkumar M, Memon N. An efficient key predistribution scheme for ad hoc network security. IEEE Journal on Selected Areas of Communication, 2005, 23(3): 611～621
63. Rivest R. The MD5 Message-Digest Algorithm. RFC 1321, IETF, April 1992
64. Eastlake D, Jones P. US Secure Hash Algorithm 1 (SHA-1). RFC 3174, IETF, September 2001
65. Zhu S, Setia S, Jajodia S. Adding reliable and self-healing key distribution to the subset difference group rekeying method for secure multicast. In: Proceedings of 5th International Workshop on Networked Group Communications. LNCS 2816, Springr, 2003. 107～118
66. Haller N. The s/key one-time password system. RFC1760, IETF, 1995
67. Perrig A, Szewczyk R, Wen V, Culler D, Tygar J D. SPINS: Security protocols for sensor networks. In: Proceedings of IEEE/ACM MobiCom, 2001. 189～199

68. Liu D, Ning P. Multilevel μTESLA: Broadcast authentication for distributed sensor networks. ACM Trans on Embedded Computing Systems, 2004, 3(4): 800～836

69. Park T, Shin K G. LiSP: A lightweight security protocol for wireless sensor networks. ACM Trans on Embedded Computing Systems, 2004, 3(3): 634～660

70. Jiang Yixin, Lin Chuang, Shi Minghui, Shen Xueming. Seal-healing group key distribution with time-limited Node Revocation for wireless sensor Networks. Ad Hoc Networks Journal, Elsevier, 2007,5(1): 14～23

71. Halevy D, Shamir A. The LSD broadcast encryption scheme. In: Proceedings of Advances in Cryptology (Crypto'02), LNCS 2442, Springer, 2002. 47～60

72. Wang P, Ning P, Reeves D S. Storage-efficient stateless group key revocation. In: proceedings of the 7th International Conference on Information Security, ISC 2004. 25～38

73. Liu D, Ning P, Sun K. Efficient self-healing group key distribution with revocation capability. In: Proceedings of the 10th ACM CCS,2003. 231～240

74. Jiang Yixin, Lin Chuang, Shen Xuemin, Shi Minghui. A Survey for key management scheme in sensor networks. Security in Sensor Networks, CRC Press. 2006

75. Wood A D, Stankovic J A. Denial of service in sensor networks. Computer, 2002, 35(10): 54～62

76. Carman D W, Kruus P S, Matt B J. Constraints and approaches for distributed sensor security. NAI Labs Technical Report, #00-010, 2000

77. Qi Hairong, et al. The development of localized algorithms in wireless sensor networks. Sensors, 2002, 2: 286～293

78. Devarapalli V, Wakikawa R, Petrescu A, Thubert P. Network Mobility (NEMO) Basic Support Protocol. RFC 3963, IETF, Jan. 2005

79. Leung K, Dommety G, Narayananand V, Petrescu A. IPv4 Network Mobility (NEMO) Basic Support Protocol. Internet Draft, draft-ietf-nemo-v4-base-00. txt. IETF, Feb. 2006

80. Ernst T. Network Mobility Support Goals and Requirements. Internet Draft, draft-ietf-nemo-requirements-05. txt, IETF, Oct. 2005

81. Ernst T, Lach H-Y. Network Mobility Support Terminology. Internet Draft, draft-ietf-nemo-terminology-05. txt. IETF, Mar. 2006

82. Thubert P, Wakikawa R, Devarapalli V. NEMO Home Network Models. Internet Draft, draft-ietf-nemo-home-network-models-06. txt. IETF, Feb. 2006

83. 任丰原,黄海宁,林闯. 无线传感器网络. 软件学报,2003, 14(7): 1282～1291

84. Warneke B, Last M, Liebowitz B, Pister KSJ. Smart dust: Communicating with a cubic-millimeter computer. IEEE Computer Magazine, 2001, 34(1): 44～51

85. Saltzer J, Reed D, Clark D. End-to-end arguments in system design. ACM Trans on Computer Systems, 1984, 2(4): 195～206

86. TinyOS. http://tinyos. millennium. berkeley. edu

87. Unmanned aerial vehicle(UAV). http://www. eecs. berkeley. edu/～pister/29Palms0103/

88. ALERT. http://www. altersystem. org

89. Bonnet P, Gehrke J, Seshadri P. Querying the physical world. IEEE Personal Communication, 2000, 7(5): 10～15

90. Noury N, Herve T, Rialle V, Virone G, Mercier E. Monitoring behavior in home using a smart fall sensor. In: Proceedings of the IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology. Lyon: IEEE Computer Society, 2000. 607～610

91. Sensor Webs. http://sensorwebs. jpl. nasa. gov/

92. Shih E, Cho S, Ickes N, Min R, Sinha A, Wang A, Chandrakasan A. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In: Proceedings of the ACM MobiCom 2001. Rome: ACM Press, 2001. 272～286

93. Asada G, Dong M, Lin T S, Newberg F, Pottie G, Kaiser W, Marcy H O. Wireless integrated network sensors (WINS) for tactical information systems. In: Proceedings of the 1998 European Solid State Circuits Conference. New York: ACM Press, 1998. 15～20

94. Heinzelman W R, Kulik J, Balakrishnan H. Adaptive protocols for information dissemination in wireless sensor networks. In: Proceedings of the ACM MobiCom'99. Seattle: ACM Press, 1999. 174～185

95. Hedetniemi S, Liestman A. A survey of gossiping and broadcasting in communication networks. Networks, 1988, 18(4): 319～349

96. Sohrabi K, Gao J, Ailawadhi V, Pottie G J. Protocols for self-organization of a wireless sensor network. IEEE Personal Communications, 2000, 7(5): 16～27

97. Estrin D, Govindan R, Heidemann J, Kumar S. Next century challenges: Scalable coordinate in sensor network. In: Proceedings of the 5th ACM/IEEE International Conference on Mobile Computing and Networking. Seattle: IEEE Computer Society, 1999. 263～270

98. Heinzelman W, Chandrakasan A, Balakrishnan H. Energy efficient communication protocol for wireless microsensor networks. In: Proceedings of the 33th Hawaii International Conference on System Sciences. Maui: IEEE Computer Society, 2000. 3005～3014

99. Manjeshwar A, Agrawal D P. TEEN: A routing protocol for enhanced efficiency in wireless sensor networks. In: Proceedings of the 15th Parallel and Distributed Processing Symposium. San Francisco: IEEE Computer Society, 2001. 2009～2015

100. Wireless sensor networks(WSN.). http://www.ececs.uc.edu/～njain/research.html

101. Lindsey S, Raghavendra C S. PEGASIS: Power-efficient gathering in sensor information systems. http://www.cs.wayne.edu/～loren/ csc8220-info/menu.html

102. Woo A, Culler D. A transmission control scheme for media access in sensor networks. In: Proceedings of the ACM MobiCom 2001. Rome: ACM Press, 2001. 221～235

103. Shih E, Cho S, Ickes N, Min R, Sinha A, et al. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In: Proceedings of the ACM MobiCom 2001. Rome: ACM Press, 272～286

104. Ye W, Heidemann J, Estrin D. An energy-efficient MAC protocol for wireless sensor network. In: Proceedings of the INFOCOM 2002. San Francisco: IEEE Computer Society, 2002

105. Singh S, Raghavendra CS. PAMAS: Power aware multi-access protocol with signaling for Ad hoc networks. ACM Computer Communication Review, 1998, 28(3): 5～26

106. Sohrabi K, Pottie G J. Performance of a novel self-organization protocol for wireless Ad hoc sensor networks. In: Proceedings of the IEEE 50th Vehicular Technology Conference. Amsterdam, 1999. 1222～1226

107. Sinhua A, Chandrakasan A. Dynamic power management in wireless sensor network. IEEE Design and Test of Computer, 2001, 18(2): 62～74

108. Lm C, Kim H, Ha S. Dynamic voltage scheduling technique for low-power multimedia application using buffers. In: Proceedings of the International Symposium on Low Power Electronics and Design. California: ACM Portal Press, 2001. 34～39

109. Elson J, Estrin D. Time synchronization for wireless sensor network. In: Proceedings of the 15th Parallel

and Distributed Processing Symposium. San Francisco: IEEE Computer Society, 2001. 1965～1970

110. Savarese C, Rabaey J. Locationing in distributed Ad-hoc wireless sensor network. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2001

111. Dynamic sensor network(DSN). http://www.east.isi.edu/projects/DSN/

112. Scalable coordination architecture for deeply distributed and dynamic system (SCADDS). http://www.isi.edu/scadds/

113. Werb J, Lanzl C. Designing a positioning system for finding things and people in indoors. IEEE Spectrum, 1998, 35(9): 71～78

114. Akyildiz F, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor network: A survey. Computer Networks, 2002, 38(4): 393～422

115. Romer K. Mattern F. The design space of wireless sensor networks. IEEE Wireless Communications, 2004, 11(6): 54～61

116. Estrin D, Govindan R, Heidermann J, Kumer S. Next century challenges: scalable coordination in sensor networks. In: Proceedings of the ACM/IEEE International conference on Mobile Computing and Networking. ACM Press, 1999. 263～270

117. GENI. Global environment for network innovations, 2006

118. 苏忠,林闯,封富君,任丰原. 无线传感器网络密钥管理的方案和协议. 软件学报,2007,18(5): 1218～1231

119. Li J Z, Li J B, Shi S F. Concepts, issues and advance of sensor networks and data management of sensor networks. Journal of Software, 2003, 14(10): 1717～1727

120. Carman D W, Kruus P S, Matt B J. Constraints and approaches for distributed sensor security. Technical Report, #00-010, NAI Lab,2000

121. Perrig A, Stankovic J, Wagner D. Security in wireless sensor networks. Communications of the ACM (Special Issue on Wireless Sensor Networks), 2004, 47(6): 53～57

122. Deng J, Han R, Mishra S. INSENS: Intrusion-tolerant routing in wireless sensor networks. Technical Report, CU-CS-939-02, Colorado University, 2002

123. Lazos L, Poovendran R. SeRLoc: Secure range-independent localization for wireless sensor networks. In: Proc of the 2004 ACM Workshop on Wireless Security. New York: ACM Press, 2004. 21～30

124. Przydatek B, Song D, Perrig A. SIA: Secure information aggregation in sensor networks. In: Proc of the 1st Int'l Conf on Embedded Networked Sensor Systems. New York: ACM Press, 2003. 255～265

125. Ye F, Luo H Y, Lu S, Zhang L X. Statistical en-route detection and filtering of injected false data in sensor networks. IEEE Journal on Selected Areas in Communications, 2005, 23(4): 839～850

126. Neuman B C, Tso T. Kerberos: An authentication service for computer networks. IEEE Communications, 1994, 32(9): 33～38

127. Mcgrew D A, Sherman A T. Key establishment in large dynamic groups using one-way function trees. IEEE Trans on Software Engineering, 2003, 29(5): 444～458

128. Basagni S, Herrin K, Bruschi D, Rosti E. Secure pebblenets. In: Proc of the 2nd ACM Int'l Symp on Mobile Ad Hoc Networking & Computing. New York: ACM Press, 2001. 156～163

129. Crossbow Technology. MICA2: Wireless measurement system

130. Shi E, Perrig A. Designing secure sensor networks. Wireless Communication Magazine, 2004, 11(6): 38～43

131. Karlof C, Sastry N, Wagner D. TinySec: A link layer security architecture for wireless sensor networks. In: Proc of the 2nd ACM Conf on Embedded Networked Sensor Systems. New York:

ACM Press，2004．162～175
132. Jiang Y X，Lin C，Shi M H，Shen X M. Security in Sensor Networks. Oxfordshire：Taylor and Francis Group，2006．113～143
133. Gaubatz G，Kaps J，Sunar B. Public keys cryptography in sensor networks—Revisited. In：Proc of the 1st European Workshop on Security in Ad-Hoc and Sensor Networks（ESAS）. New York：ACM Press，2004．2～18
134. Malan D J，Welsh M，Smith M D. A public-key infrastructure for key distribution in TinyOS based on elliptic curve cryptography. In：Proc of the 1st IEEE Int'l Conf on Sensor and Ad Hoc Communications and Networks. IEEE Press，2004．71～80
135. Eschenauer L，Gligor V. A key management scheme for distributed sensor networks. In：Proc of the 9th ACM Conf on Computer and Communications Security. New York：ACM Press，2002．41～47
136. Chan H，Perrig A，Song D. Random key predistribution schemes for sensor networks. In：Proc of the 2003 IEEE Symp on Security and Privacy. Washington：IEEE Computer Society，2003．197～213
137. Du W，Deng J，Han Y S，Varshney P K. A pairwise key pre-distribution scheme for wireless sensor networks. In：Proc of the 10th ACM Conf on Computer and Communications Security. New York：ACM Press，2003．42～51
138. Liu D，Ning P. Establishing pairwise keys in distributed sensor networks. In：Proc of the 10th ACM Conf on Computer and Communications Security. New York：ACM Press，2003．52～61
139. Liu D，Ning P. Location-based pairwise key establishments for static sensor networks. In：Proc of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks. New York：ACM Press，2003．72～82
140. Du W，Deng J，Han Y S，Chen S，Varshney P K. A key management scheme for wireless sensor networks using deployment knowledge. In：Proc of the IEEE INFOCOM. Piscataway：IEEE Press，2004．586～597
141. Huang D，Mehta M，Medhi D，Harn L. Location-aware key management scheme for wireless sensor networks. In：Proc of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks. New York：ACM Press，2004．29～42
142. Chan H，Perrig A. PIKE：Peer intermediaries for key establishment in sensor networks. In：Proc of the IEEE INFOCOM 2005. Piscataway：IEEE Communication Society，2005．524～535
143. Camtepe S A，Yener B. Combinatorial design of key distribution mechanisms for wireless sensor networks. In：Proc of the Computer Security—ESORICS. Berlin：Springer-Verlag，2004．293～308
144. Perrig A，Szewczyk R，Tygar J，Wen V，Culler D. SPINS：Security protocols for sensor networks. ACM Wireless Network，2002，8(5)：521～534
145. Zhu S，Setia S，Jajodia S. LEAP：Efficient security mechanisms for large-scale distributed sensor networks. In：Proc of the 10th ACM Conf on Computer and Communications Security. New York：ACM Press，2003．62～72
146. Younis M，Ghumman K，Eltoweissy M. Location-aware combinatorial key management scheme for clustered sensor networks. IEEE Trans on Parallel and Distribution System，2006，17(8)：865～882
147. Eltoweissy M，Moharrum M，Mukkamala R. Dynamic key management in sensor networks. IEEE Communications Magazine，2006，44(4)：122～130
148. Moharrum M A，Eltoweissy M. A study of static versus dynamic keying schemes in sensor networks. In：Proc of the 2nd ACM Int'l Workshop on Performance Evaluation of Wireless Ad Hoc，Sensor，and Ubiquitous Networks. New York：ACM Press，2005．122～129

149. Blundo C, Santis A D, Herzberg A, Kutten S, Vaccaro U, Yung M. Perfectly secure key distribution for dynamic conferences. Information and Computation, 1998, 146(1): 1～23

150. Bollobás B, Fulton W, Katok A, Kirwan F, Sarnak P. Rand Graphs. 2nd edn. Cambridge: Cambridge University Press, 2001. 160～200

151. Blom R. An optimal class of symmetric key generation systems. In: Beth T, Cot N, Ingemarsson I, eds. Proc of the Eurocrypt'84. New York: Springer-Verlag, 1984. 335～338

152. Liu D, Ning P. Multilevel μTESLA: Broadcast authentication for distributed sensor networks. ACM Trans. on Embedded Computing Systems, 2004, 3(4): 800～836

153. Liu D, Ning P, Zhu S, Jajodia S. Practical broadcast authentication in sensor networks. In: Proc of the 2nd Annual Int'l Conf on Mobile and Ubiquitous Systems: Networking and Services. Washington: IEEE Computer Society, 2005. 118～129

154. Eltoweissy M, Heydari H, Morales L, Sudborough H. Combinatorial optimization of key management in group communications. Journal of Network and Systems Management, 2004, 12(1): 33～50

155. Huang Q, Cukier J, Kobayashi H. Liu B, Zhang J. Fast authenticated key establishment protocols for self-organizing sensor networks. In: Proc of the 2nd ACM Int'l Conf on Wireless Sensor Networks and Applications. New York: ACM Press, 2003. 141～150

156. Kotzanikolaou P, Magkos E, Douligeris C, Chrissikopoulos V. Hybrid key establishment for multiphase self-organized sensor networks. In: Proc of the 6th IEEE Int'l Symp on a World of Wireless Mobile and Multimedia Networks. Washington: IEEE Computer Society, 2005. 581～587

157. Zhang Y C, Liu W, Lou W J, Fang Y G. Location-based compromise-tolerant security mechanisms for wireless sensor networks. IEEE Journal on Selected Areas in Communications, 2006, 24(2): 247～260

158. Shamir A. Identity based cryptosystems and signatures schemes. In: Proc of the Advances in Cryptology. New York: Springer- Verlag, 1984. 47～53

159. Pietro R D, Mancini L V, Mei A, Panconesi A, Radhakrishnan J. Connectivity properties of secure wireless sensor networks. In: Proc of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks. New York: ACM Press, 2004. 53～58

160. Hwang J, Kim Y. Revisiting random key pre-distribution schemes for wireless sensor networks. In: Proc of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks. New York: ACM Press, 2004. 43～52

161. Chan H, Gligor V D, Perrig A, Muralidharan G. On the distribution and revocation of cryptographic keys in sensor networks. IEEE Trans on Dependable and Secure Computing, 2005, 2(3): 233～247

162. Wood A D, Stankovic J A. Denial of service in sensor networks. Computer, 2002, 35(10): 54～62

163. Zhu S, Setia S, Jajodia S, Ning P. An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks. In: Proc of the IEEE Symp on Security and Privacy. Oakland: IEEE Computer Society, 2004. 259～271

164. Marti S, Giuli T J, Lai K, Baker M. Mitigating routing misbehavior in mobile ad hoc networks. In: Proc of the 6th Annual Int'l Conf on Mobile Computing and Networking. New York: ACM Press, 2000. 255～265

165. Staddon J, Miner S, Franklin M, et al. Self-Healing key distribution with revocation. In: proceedings of the IEEE Symp. on Security and Privacy. IEEE Computer Society, 2002. 241～257

Chapter
第 5 章

基于应用层组播的视频安全

随着计算机网络技术的飞速发展，能将多媒体信息同时传输给接入方式不同、计算能力各异、质量需求不同的终端系统的数字服务已日益成为市场需求的热点，如付费观看（pay-per-view）、视频会议、视频点播（video on demand）、军事指挥控制、在线电视和联网游戏等。组播（multicast）通信方式由于在进行一对多通信时能够有效地降低骨干网上冗余数据包的传输数量，所以在多媒体通信领域得到了深入的研究。传统的组播方式是网络层组播（IP multicast），尽管从 IETF（Internet Engineering Task Force）提出关于网络层组播的 RFC（request for comments）已有 20 余年，但是网络层组播由于可扩展性差、缺乏拥塞控制、难以管理、部署难度大等问题，一直得不到大规模的应用。应用层组播是由参与服务的主机构成组播树结构，组播功能在应用层实现的通信模式，因此应用层组播继承了组播模式的通信效率，克服了 IP 层组播难以在 Internet 中应用的缺点。基于应用层组播的大规模流媒体传输体系近年来成为了研究热点和多媒体应用的发展方向。

本章首先对数字水印进行介绍，给出了数字水印典型算法；讨论了基于视频的可靠密钥嵌入算法和选择性加密算法，然后提出了一个媒体相关的视频安全组播协议，给出了实验与分析结果。最后，阐述了视频流传输的差错控制机制，介绍了 MPEG-4 编码标准、信源差错控制编码和信道差错控制编码等，并针对无线网络，提出一个动态优化组包策略，适用于各种视频编码算法，且增强了抗丢包能力。

5.1 国内外研究现状和进展

基于网络层的 IP 组播，由于其对网络层的依赖性，实现必须得到路由器的支持，因此没有得到广泛的应用。随着 Internet 的发展和传输能力的提高，应用层组播成为 IP 组播的有力竞争者。应用层组播使用网络层的单播服务，依靠参与者之间的协作来发送和分布组播信息。当前，基于应用层组播的流媒体系统研究工作已有不少的成果，如 NICE[1]，Narada[2]，HMTP[3]，OMNI[4]，Spread It[5]，CoopNet[6] 和 SplitSream[7] 等。一些系统已在 Internet 中应用，如 2003 年 SIGCOMM 会议曾利用“端系统组播（end system multicast）”系统进行现场直播，位于全球不同位置的用户运行该系统通过 Internet 可观看此次会议实况。

与此同时,伴随着人们对多媒体业务需求的日益增长,多媒体信息的数字版权保护问题逐渐成为各项多媒体网络服务及应用进一步发展的关键[8,9]。如在收费电视方面,目前国外许多有线电视和卫星电视已从纯商业广告经营方式转向采用节目分层次付费的营运管理方式,并已取得巨大成功,这很大程度上依赖于有效的版权保护机制来保障运营者的利益。但是,由于流媒体具有在线播放的特点,而应用层组播技术也有很多不利于安全传输的特点,因此对基于应用层组播的流媒体系统进行版权保护面临着很大的困难和挑战。

基于应用层组播的大规模流媒体应用中的版权保护机制主要解决一个问题,即防止非法用户对受版权保护的多媒体信息的访问。这个问题实际上包含了两个子问题:一是如何保证只有合法的用户才能解码出组播中的多媒体信息,实际上是多媒体数据的安全传输问题,即多媒体信息的加密与组播中的密钥分发问题;二是如何避免合法用户非法复制受版权保护的多媒体信息,即如何利用数字水印技术来有效保护媒体著作权人的合法利益,避免非法盗版的威胁。

对于第二个问题,即通过采用数字水印技术来实现版权保护的功能,目前已有很多研究成果[10],而且该部分功能比较独立,完全可以直接采用现有的解决方案。另外,本章内容主要针对流媒体应用中的视频传输,由于音频部分是和视频同步的,可采用与视频相似的方法保证其传输安全,所以也不作重点研究。因此,本章将以应用层组播中的视频安全传输机制作为主要的研究对象。

现有的针对大规模流媒体组播中的视频安全传输机制的基本解决方案是对组播的视频信息进行加密,而加密和解密用的密钥只有通过了身份认证的组成员才知道,这个密钥被称为会话密钥(session key,SK)。但是现有的研究主要针对网络层组播的应用而提出的,针对应用层组播的应用还没有很好的方案。同时考虑到流媒体业务对 QoS(服务质量)保障的高要求,如何在取得版权保障的同时,尽量减少对多媒体通信中服务质量的影响,也是一个非常重要的问题。随着各种应用技术的发展,信息安全和网络安全的形势异常严峻,安全协议日趋复杂,相应地,对带宽与计算资源的消耗也不断增加,如何在安全性和流媒体运行效率之间做出一个很好的权衡自然成为又一个需要解决的问题。

归纳起来,如果要有效地解决这一安全传输机制的问题,必须依赖如下 3 个关键技术的支持:①高可扩展的密钥管理机制;②视频加密算法;③用户的身份认证机制。其中,高可扩展的密钥管理机制主要研究的是在应用层组播中如何安全、高效地分发、管理与更新会话密钥,由于应用层组播是通过网络终端对数据包的复制来实现组播的,所以应用层组播在拥有较多优势的同时,也具有如下缺点:可靠性差、延迟比较大、传输效率不如 IP 组播[11]。在这种情况下,设计相关的密钥管理机制面临着很大的挑战。对视频加密算法而言,考虑到视频通信中一定的特殊性:①视频流传输的数据量非常大,需要加密算法比较有效,同时,大数据量也为入侵者进行统计攻击提供了条件,需要加密算法能够应对统计攻击;②视频传输流中的重要信息一般不多,即使丢失或改动某些信息,人眼在接收端也很难识别出来或者也能接受;③视频通信对时延的敏感性;④对安全性的要求相对数据通信效率而言要低,要求视频加密算法要同时兼顾实时性与安全性的需要。用户的身份认证机制要确保用户身份的合法性,并确保其组播信息的不可否认性。目前,比较有代表性的身份验证方法是基于权威机构颁发的公钥证书的 PKI 认证方式[12]。其他简单的认证方式,如基于预先共享密钥的认证方式等。由于在该领域的研究相对比较独立与成熟,所以这里不对该问题进行专门

研究，而是直接采用已有的方式来验证用户的身份。

我们将相关的研究按密钥分发信道的不同分为两类[8,13]：媒体无关的密钥分发解决方案（media-independent approach）和媒体相关的密钥分发解决方案（media-dependent approach）。

在媒体无关的密钥分发方案中，密钥的更新信息通过独立于媒体内容的信道传输给组播成员。密钥的分发通常要使用可靠组播协议来分发。当用媒体无关的密钥分发机制进行密钥分发时，一个很重要的弱点是容易受到攻击，在网上的非法监听人员可以从该通道获取很多重要信息来攻击该系统，如可以知道组播成员的数量等，文献[14]分析了相关系统的弱点。同时，由于该方式没有考虑到流媒体的特性，也无法有效支持现有的服务质量控制策略，因而大大降低了流媒体通信中的服务质量。

在媒体相关的密钥分发机制中，密钥被嵌入到多媒体信息中，与多媒体信息中的内容一起通过一个信道发送给组播成员。密钥嵌入技术与数字水印技术有很多相似的地方，都是利用视频编码中的一些特点，如 DCT（discrete cosine transform，离散余弦变换）系数块、运动估计的精度等来隐藏密钥。当然两者也有所不同，如密钥信息比水印信息要少，而对传输的可靠性要求很高。总的来说，媒体相关的密钥分发方案相对媒体无关的密钥分发方案来说有很多的优点：

（1）通过将要更新的密钥嵌入到多媒体信息中来分发，可以有效地降低密钥更新时对网络带宽等资源的消耗。

（2）密钥嵌入到多媒体信息中并且随多媒体信息一起发送，网上的非法监听人员无法监听到密钥更新的信息。

（3）有效提高了安全性，因为监听者要破解密钥，不仅需要攻击会话密钥（SK）和用于加密 SK 的密钥（KEK），还必须知道密钥的内嵌规则，这增大了攻击的难度，因此增强了密钥的安全性。

（4）使用媒体相关通道进行密钥传输，可以通过应用层重传机制和差错控制来保证传递密钥重新分发消息的可靠性；视频编码中诸多的差错控制算法也都可以被用来提高密钥分发的可靠性。

（5）对于可伸缩的视频传输系统而言，一个视频源可能会提供多层不同精度的视频，现有的许多应用层分层组播系统属于这种情况，对这种情况而言，使用媒体相关通道进行密钥传输是最适合的。因为若用媒体无关的密钥分发机制进行密钥分发的话，则系统的开销和成本太大了。

由于媒体相关的密钥分发方式具有性能和安全方面的双重优势，所以在本章中采用媒体相关的密钥分发方案。用媒体相关的分发机制来解决基于应用层组播的视频安全传输机制的问题，需要深入研究以下内容：①基于视频的密钥嵌入算法的研究；②视频加密算法的研究；③密钥管理与分发机制的研究；④视频流传输的差错控制研究。对媒体相关的方案和媒体无关的方案而言，区别在于传输媒介不同，但是对密钥更新与分发的策略而言，基于两者的方法区别不大。

5.2 流媒体与应用层组播概述

5.2.1 流媒体技术

流媒体技术广泛用于在线直播、视频点播、远程教育、多媒体新闻发布、网络广告、电子商务、远程医疗、网络电台、实时视频会议等互联网信息服务的方方面面，它的应用将为网络信息交流带来革命性的变化，将对人们的工作和生活产生深远的影响。

5.2.1.1 流媒体定义

流媒体(streaming media)是指在网络上按时间先后次序传输和播放的连续音/视频数据流。流媒体把连续的影像和声音信息经过特殊的压缩方式分成一个个压缩包，由音/视频服务器向用户计算机连续、实时地传送。让用户一边下载一边观看和收听，不需要将整个压缩文件下载到自己的机器后才可以观看。该技术先在用户端的电脑上创建一个缓冲区，在播放前预先下载文件的一小段数据作为缓冲，播放程序取用这一小段缓冲区内的数据进行播放。在播放的同时，多媒体文件的剩余部分在后台继续下载填充到缓冲区。这样，当网络实际连接速度大于播放所消耗数据的速度时，就可以避免播放的中断，也使得播放品质得以维持。流媒体数据流具有 3 个特点：连续性、实时性和时序性，即其数据流具有严格的前后时序关系。

与传统多媒体相比，流媒体具有以下优点：

(1) 启动延迟大幅度地缩短，用户不用等待所有内容下载到硬盘上才开始浏览/观看。一般来说，一个 45 分钟的影片片段在 1 分钟内就能够显示在客户端上，而且在播放过程中一般不会出现断续的情况。

(2) 对系统缓存容量的需求大为降低。由于 Internet 是以保证传输为基础进行断续的异步传输，数据被分解成许多包进行传输，动态变化的网络使各个包可能选择不同的路由，故到达用户计算机的时间延迟也就有所不同。所以，在客户端需要缓存系统来弥补延迟和抖动的影响以及保证数据包传输顺序的正确性，使媒体数据能够连续输出，不会因网络暂时拥堵而使播放出现停顿。虽然流式传输仍需要缓存，但由于不需要把多媒体文件所有的动画、音/视频内容都下载到缓存中，因此，对缓存的要求大为降低。

(3) 流式传输的实现有特定的实时传输协议，更加适合动画、音/视频在网上的流式实时传输。

5.2.1.2 流式传输

流式传输是流媒体实现的关键技术，其定义很广泛，现在主要是指通过网络传送媒体的技术总称。首先，流式传输的实现需要缓存。通常高速缓存所需容量并不大，因为高速缓存使用环形链表结构来存储数据，通过丢弃已经播放的内容，流可以重新利用空出的高速缓存空间来缓存后续尚未播放的内容。

其次，流式传输的实现需要合适的传输协议。WWW技术是以HTTP为基础的，而HTTP建立在TCP的基础上。由于TCP需要较多的开销，故不太适合传输实时数据。在流式传输的实现方案中，一般采用HTTP/TCP来传输控制信息，而用RTP(real-time transport protocol，实时传输协议)/UDP来传输实时音/视频数据。

流式传输的过程一般是：用户选择某一流媒体服务后，Web浏览器与Web服务器之间使用HTTP/TCP交换控制信息，以便把需要传输的实时数据从原始信息中检索出来；然后客户机上的Web浏览器启动流媒体播放程序，使用HTTP从Web服务器检索相关参数初始化流媒体播放程序。这些参数可能包括目录信息、A/V数据的编码类型或与A/V检索相关的服务器地址。流式传输的过程如图5.2.1所示。

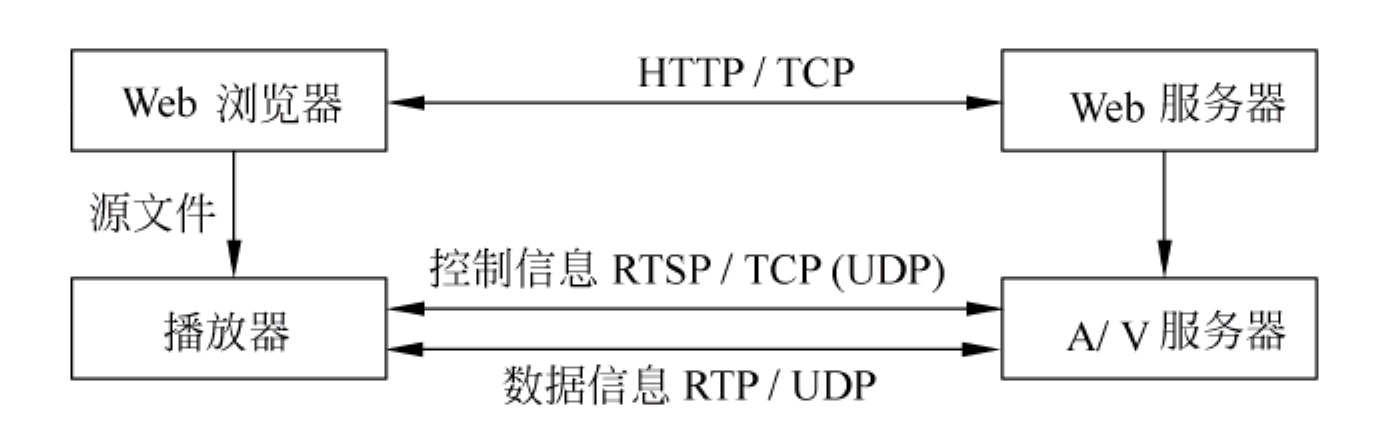


图5.2.1　流式传输过程

实现流式传输有两种方法：实时流式(real-time streaming)传输和顺序流式(progressive streaming)传输。一般来说，如视频为实时广播，使用流式传输媒体服务器，即为实时流式传输。如使用HTTP服务器，文件即通过顺序流发送，则称为顺序流式传输。采用哪种传输方法依赖于用户的具体需求，当然，流式文件也支持播放前完全下载到硬盘后再播放的方式。

(1) 顺序流式传输

顺序流式传输是顺序下载，用户可以观看在线媒体。但在给定时刻，用户只能观看已下载的那部分，而不能跳到未下载的前序部分；不能根据用户的连接速度作调整。由于标准的HTTP服务器可发送这种形式的文件，而不需要其他特殊协议，所以经常称之为HTTP流式传输。

顺序流式传输方式适合高质量的短片段，如片头、片尾和广告，由于文件在播放前观看的部分是无损下载的，所以这种方法能够保证电影播放的最终质量。顺序流式文件放在标准HTTP或FTP服务器上，易于管理，基本上与防火墙无关。顺序流式传输不适合长片段和有随机访问要求的视频、讲座、演说与演示，也不支持现场广播，严格地说，它是一种点播技术。

(2) 实时流式传输

实时流式传输是指保证媒体信号带宽与网络连接相匹配，使媒体可被实时地观看。实时流与HTTP流式传输不同，需要专用的流媒体服务器与传输协议。实时流式传输是实时传送，特别适合现场事件，也支持随机访问，用户可快进或后退以观看前面或后面的内容。理论上，实时流一经播放就可不停地收看，但实际上，可能会发生周期暂停。从视频质量上讲，实时流式传输必须匹配连接带宽，由于出错丢失的信息被忽略掉，网络拥挤或出现问题时，视频质量会很差，如要保证视频质量，顺序流式传输效果更好。

实时流式传输需要特定服务器，如 QuickTime Streaming Server，Real Server 与 Windows Media Server，这些服务器允许对媒体发送进行更多级别的控制，因而系统设置、管理比标准 HTTP 服务器更加复杂。

5.2.1.3 流媒体实现原理

流媒体实现原理简单来说，就是通过采用高效的压缩算法，在降低文件大小的同时伴随质量的损失，让原有的庞大的多媒体数据适合流式传输。然后通过架设流媒体服务器，修改相应的标识，利用各种实时协议传输流数据。流媒体实现原理如图 5.2.2 所示。

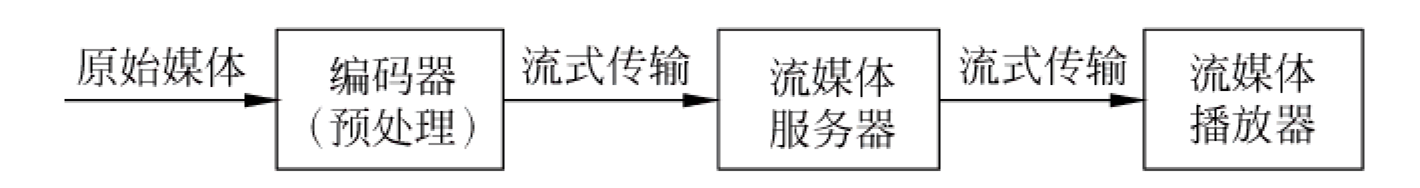


图 5.2.2 流媒体实现原理

1. 预处理

多媒体数据必须进行预处理才能适合流式传输，这是因为目前的网络带宽相对多媒体巨大的数据流量来说还显得远远不够。预处理主要包括两个方面：一是采用先进、高效的压缩算法；二是加入一些附加信息把压缩媒体转为适合流式传输的文件格式。其关键在于压缩原始的音/视频内容，使其能够在窄带或宽带通道上以流的方式传给用户。预处理在编码器内完成，编码方式的选择可以是多种多样的。

音/视频编码器在功能上有相当大的差别。最终的编码资料可以是利用文本、图形、脚本形式进行多路传输的，并且是放在能够实现流的方式的文件结构中，也就意味着，该文件由时间标记以及其他易于实现流的方式的特点，然后再在客户端进行解码。编码过程应该考虑不同编码速度的定制性能、包损失的容错性与网络的带宽波动、最低速度下好的音/视频品质、编码/流式传送的成本、流的控制及其他方面等。

2. 媒体服务器

流媒体系统中的媒体服务器用于存放和控制流媒体的数据。随着流媒体规模的扩大，流媒体服务器的性能成为制约流媒体服务扩展能力的重要因素。流媒体服务器性能的关键指标是流输出能力以及能同时支持的并发请求数量。影响流媒体服务器性能的因素很多，包括 CPU 能力、I/O 总线、存储带宽等。通常单个流媒体服务器的并发数都在几百以内，因此为了具有更好的性能，目前的高性能流媒体服务器都采用大规模并行处理的结构，例如采用超立方体的结构将各个流媒体服务单元连接起来。还有一种方法是采用简单的 PC 集群的方式，这种方式下多个 PC 流媒体服务器用局域网连接，前端采用内容交换/负载均衡器将流媒体服务的请求分布到各个 PC 媒体服务单元。后一种方式的性能不如前一种方式，但是成本低，容易实现。

3. 流媒体传输协议

流式传输的实现需要合适的传输协议。TCP 需要较多的开销，故不太适合传输实时数据。在流式传输的实现方案中，一般采用 HTTP/TCP 来传输控制信息，而用 RTP/UDP 来

传输实时多媒体数据。

（1）实时传输协议 RTP 与 RTCP

RTP(Real-Time Transport Protocol,实时传输协议)是用于 Internet/Intranet 针对多媒体数据流的一种传输协议。RT 被定义在一对一或一对多传输的情况下工作,其目的是提供时间信息和实现流同步。RTP 通常使 UDP 来传送数据,但 RTP 也可以在 TCP 或 ATM 等其他协议上工作。当应用程序开始一个 RTP 会话时将使用两个端口：一个给 RTP,另一个给 RTCP(Real-time Transport Control Protocol,实时传输控制协议)。RTP 本身并不能为顺序传送数据包提供可靠的传送机制,也不提供流量控制或拥塞控制,它依靠 RTCP 提供这些服务。RTCP 和 RTP 一起提供流量控制和拥塞控制服务。RTP 和 RTCP 配合使用,它们能以有效的反馈和最小的开销使传输效率最佳,因而特别适合传送网上的实时数据。

（2）实时流协议 RTSP

RTSP(Real-Time Streaming Protocol,实时流协议）是由 RealNetworks 和 Netscape 共同提出来的,该协议定义了一对多应用程序如何有效地通过 IP 网络传送多媒体数据。RTSP 在体系结构上位于 RTP 和 RTCP 之上,它使用 TCP 或 RTP 完成数据传输。HTTP 与 RTSP 相比,HTTP 传送 HTML,而 RTP 传送的是多媒体数据。HTTP 请求由客户机发出,服务器作出响应;使用 RTSP 时,客户机和服务器都可以发出请求,即 RTSP 可以是双向的。

（3）资源预留协议 RSVP

由于音频和视频数据流比传统数据对网络的延时更敏感,要在网络中传输高质量的音频、视频信息,除带宽要求之外,还需要其他更多的条件。RSVP(Resource Reserve Protocol)是 Internet 上的资源预留协议,使用 RSVP 预留一部分网络资源,能在一定程度上为流媒体的传输提供 QoS 保障。

5.2.2　应用层组播技术

随着 Internet 的不断发展,网络用户的大量增加,各种多媒体业务的大量应用以及越来越多的新兴业务的涌现,例如视频点播、电视电话会议、远程教学等,导致传统的点到点的单播通信方式由于其严重的带宽浪费和效率低已经不能适应这些要求了,于是人们提出了组播的概念。组播是一种通过单一的发送操作将数据报从一点传送到多点的通信方式。

目前应用层组播研究集中于视频会议系统、媒体流的分发系统(如视频广播）和订阅(publish)/分发系统(subscribe system）等,它主要用于实时的多媒体传输。这里利用了多媒体信息的性质,即在传输链路质量下降时,用户仍可利用收到的低速率或者不完整的信息,也利用了组播“时间上集中、空间上分布”的特点。

5.2.2.1　应用层组播原理

应用层组播(application layer multicast,ALM)的基本思想是将组成员组织成一个覆

盖网络(overlay network),通过组成员之间的协作实现高效、可靠的数据传输服务。应用层组播将路由器的组播功能提升至端主机的应用层实现,即组成员主机在接收报文的同时,还将报文复制并传递给其他组成员主机,实现了应用层的数据组播(报文在网络层实际是用单播机制传送的)。通过应用层组播,所有加到系统的节点不仅使用其他节点提供的流媒体服务,而且同时为其他节点提供服务,使实现轻量级的流媒体服务器成为可能。

并不像 IP 组播那样,数据报在路由器进行复制转发,应用层组播的数据在端系统上复制并转发,同时端系统也负责组播组成员的管理。端系统在逻辑上组成了一个叠加网络拓扑,而应用层组播协议的主要目标就是构造并维护一个有效的数据传输拓扑。图 5.2.3 对简单单播、组播和应用层组播作了一个比较。

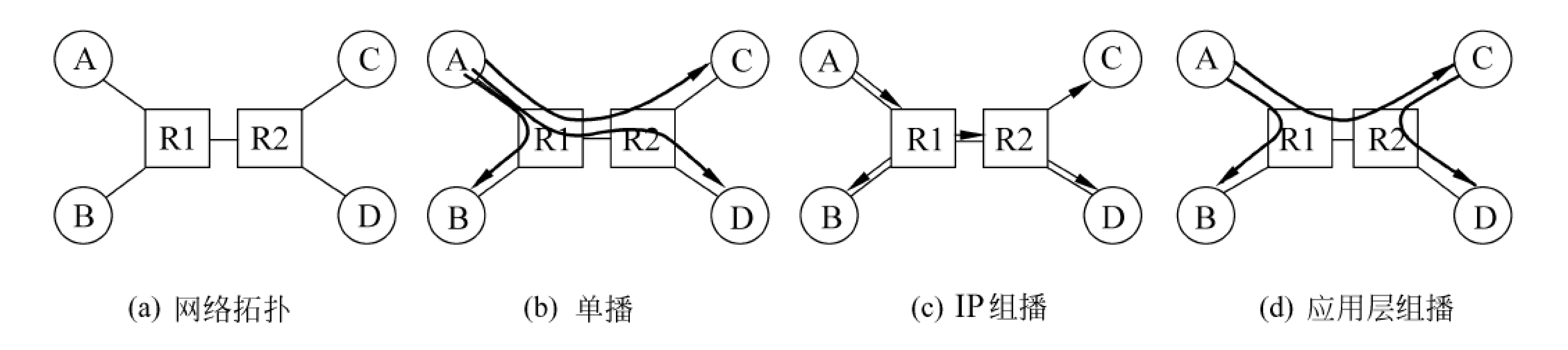


图 5.2.3　1 单播、IP 组播和应用层组播

图 5.2.3(b)描述了简单的单播传输。可以看出,在靠近数据源的路径上出现了非常大的数据传输冗余(例如,链路 A-R1 会传输 3 份相同的数据传输),并因此导致在花费较高的链接(R1-R2)上出现重复的数据传输。

图 5.2.3(c)描述了用 DVMRP(distance vector multicast routing protocol,距离向量组播路由协议)构造的一个 IP 组播树。可以看到,IP 组播有效地避免了冗余的数据传输,并且在每一个物理链接上仅有一个数据传输,同时也使每个接收者获得了与单播同样的低延时。

图 5.2.3 (d)描述了用应用层组播协议构成的一个叠加组播树,与图 5.2.3(b)所用的简单单播技术相比,临近数据源 A 的链路 A-R1 上只出现 2 次相同的数据传输,同时在花费较高的链路 R1-R2 上并没有出现数据传输冗余,由此可以看出,不需要任何的网络底层改变,也可以实现比较有效的组播传输。

5222　应用层组播的优缺点

端系统实现的组播功能可以避免网络层实现组播的许多难题,其优点如下:

(1) 应用层组播的状态在主机系统中维护,不需要路由器保持组的状态,即不需要改变现有网络路由器,解决了业务的扩展问题,使得网络可以支持大量的组播组。

(2) 接入控制更容易实现。由于单播技术在这方面比较成熟,而应用层组播是通过终端系统之间单播来实现的,所以差错控制、流控制、拥塞控制容易实现。

(3) 无 IP 组播中复杂的地址结构和复杂的协议。

应用层组播也存在以下缺点:

(1) 可靠性相对较差:终端系统的可靠性比路由器要差。

(2) 可扩展性不好：底层的路由信息对应用层组播来说是隐藏起来的，可扩展性不好。

(3) 延迟比较大：IP 组播主要是链路上的延迟，而在应用层组播中，数据还要经过终端系统，因而延迟相对要大一点。

(4) 传输效率不如 IP 组播：应用层组播在数据传输过程中会产生数据冗余，因此它们比 IP 组播的效率要差。

5.2.2.3　应用层组播与 IP 组播的区别

IP 组播是对互联网的“单播、尽力转发”模型的重要扩充，组播的主要功能在路由器上实现，通过合并重复信息传输来减少带宽浪费和降低服务器的负担。IP 组播的主要思想是，在互联网单播的框架上进行扩展，功能主要通过路由器来实现。组播适用于那些在时间上具有集中性、而在空间上具有分布性的应用。IP 组播适用于实时、不可靠的应用。

由于能够有效地减少数据包的复制到最小的限度，IP 组播一直以来被认为是一种最有效的实现数据的群分发的方法，但是由于 IP 组播在传输技术和管理上的缺点，使得 IP 组播至今并没有能够得到广泛的应用。

(1) IP 组播要求路由器为每一个组播组保留状态信息，可扩展性较差。这样，路由器的路由和转发表将需要对每一个不同的组播地址保留一个相应的路由表项，但是组播地址并不像单播地址那样容易集成，因此增加了路由器的系统开销和复杂性。

(2) IP 组播是一种尽力而为的服务。当要提供高层的特性时，如可靠传输、拥塞控制、流量控制以及安全管理等，会比简单的单播更困难，因此 Internet 服务提供商不愿意提供 IP 组播的支持。虽然目前已经出现了针对上面这些特性的研究，但是这些解决方案目前在 Internet 上的影响并不明确，需要在大范围应用前进行更好的研究。

(3) IP 组播需要对现有网络作底层的改变。同时，由于在收费机制方面的技术无法突破，使得目前只有少数的 Internet 服务提供商支持 IP 组播。

出于上述考虑，国外一些研究者开始研究新的组播架构，围绕 IP 组播的种种难题，提出了基于应用层的组播协议，在应用层实现组播的功能，不再依靠网络层路由器来实现。这种组播方法不需要任何网络底层架构的改变来实现组播，从而为组播的大范围开展与应用提出了一条新的途径。

应用层组播与 IP 组播的主要区别如下：

(1) 报文转发位置不同。应用层组播数据转发节点是覆盖式网络中的终端主机，而 IP 组播的报文转发必须由核心路由器来处理。

(2) 网络拓扑的创建方法不同。应用层组播的覆盖式网络是由节点间直连而成的一个虚拟图(有向图或无向图)，完全隐藏了底层的物理网络拓扑。这种覆盖式网络拓扑是完全可控的，且可以利用一些额外的知识或特定的度量对网络拓扑进行优化。而在 IP 组播中，路由器是预先部署的，因此网络拓扑难以控制和改变。

(3) 组成员关系维护方式不同。IP 组播的组成员关系信息分布于组播路由器，而应用层组播的成员关系由系统中的汇聚点集中控制或完全分散于各个节点。

5.3 数字水印

随着数字技术和Internet的发展,图像、音频、视频等形式的多媒体数字作品纷纷在网络上发布,其版权保护与信息完整性保证逐渐成为迫切需要解决的一个重要问题。数字水印(digital watermarking)是近几年来国际学术界兴起的一个前沿研究领域,作为一种信息隐藏技术,广泛应用于媒体版权保护等方面。数字水印作为信息隐藏技术研究领域的重要分支,是实现多媒体版权保护与信息完整性保证的有效方法,目前也正成为信息领域的一个研究热点。其主要思想是在数字媒体内容之中嵌入某种信息(即数字水印)来证实该数据的所有权。在发生数字媒体侵权使用、版权争议时,通过检测媒体内容中的数字水印判断其来源,从而起到对多媒体信息进行版权保护的目的。被嵌入的水印可以是一段文字、标识、序列号或者是一幅图像等,而且通常是不可见的,它与原始数据紧密结合并隐藏在其中,并且可以经历一些不破坏原始数据使用价值或商用价值的操作而得以保存下来。

5.3.1 数字水印的特点及应用

在数字水印系统中,水印信息的丢失,就意味着版权保护信息的丢失,从而也就失去了版权保护的功能。因此水印信息应尽可能地隐蔽不易被发现,同时还必须考虑水印数据在经历各种正常和非正常的操作之后仍具有免遭破坏的能力,因此数字水印技术必须具有如下特性:

(1) 不可见性(imperceptibility):数字水印的嵌入不应使得原始数据发生可感知的改变,也不能使得被保护数据在质量上发生可以感觉到的失真。

(2) 鲁棒性(robustness):当被保护的数据经过某种改动或者攻击(如传输、编码、有损压缩等)以后,嵌入的水印信息应保持一定的完整性,并能以一定的正确概率被检测到。

(3) 安全性(security):数字水印应该难以被伪造或者加工,并且未经授权的个体不得阅读和修改水印,理想情况是未经授权的客户将不能检测到产品中是否有水印存在。

(4) 可证明性:在实际的应用过程中,可能多次加入水印,那么数字水印技术必须能够允许多重水印嵌入被保护的数据,且每个水印均能独立地被证明。

数字水印主要应用在以下几个方面[15~23]:

(1) 版权保护:数字作品的所有者可用密钥产生水印,并将其嵌入原始数据,然后公开发布其水印版本作品。当该作品被盗版或出现版权纠纷时,所有者即可从被盗版作品中获取水印信号作为依据,从而保护其合法权益。

(2) 数字指纹:为避免数字作品未经授权被复制和发行,版权所有人可以向分发给不同用户的作品中嵌入不同的水印以标识用户的信息。该水印可根据用户的序号和相关的信息生成,一旦发现未经授权的复制,就可以根据此复制所恢复出的指纹来确定它的来源。

(3) 认证和完整性校验:通常采用脆弱水印。对插入了水印的数字内容进行检验时,需用唯一的与数据内容相关的密钥提取出水印,然后通过检验提取出的水印完整性来检验数字内容的完整性。一旦发现水印遭到破坏,就表示内容已经被改动过了。其优点在于,认

证同内容密不可分，因此简化了处理过程。

（4）访问控制：利用数字水印技术可以将访问控制信息嵌入到媒体中，在使用媒体之前通过检测嵌入到其中的访问控制信息，以达到访问控制的目的，它要求水印具有很高的鲁棒性。

（5）信息隐藏：数字水印可用于作品的标识、注释、检索信息等内容的隐藏，这样不需要额外的带宽，且不易丢失。另外，数字水印技术还可以用于隐蔽通信，这将在国防和情报部门得到广泛的应用。

5.3.2　数字水印的基本原理和评价标准

1. 常规的嵌入检测框架

图 5.3.1 所表示的是常规水印嵌入模型，其功能是根据密钥 KEY 生成水印信号 W，通过一定的方法加入原始数据中，得到嵌入了水印的作品。在水印信号生成过程中，通常是需要原始数据的，其作用是使生成的水印信号与原始数据相关，即在不同的数据中嵌入的水印信号各不相同。图 5.3.2 是常规的水印检测模型，其功能是根据 KEY 生成水印信号 W，然后与待测数据进行水印信号相似性检测，判断是否存在水印。生成水印信号是否使用待测数据需与水印嵌入过程中的生成方法一致。一些水印技术（如私有水印）中，检测过程需要使用原始数据，以便有效解决一些水印鲁棒性问题，但这同时也带来了一些额外的开销和安全隐患。

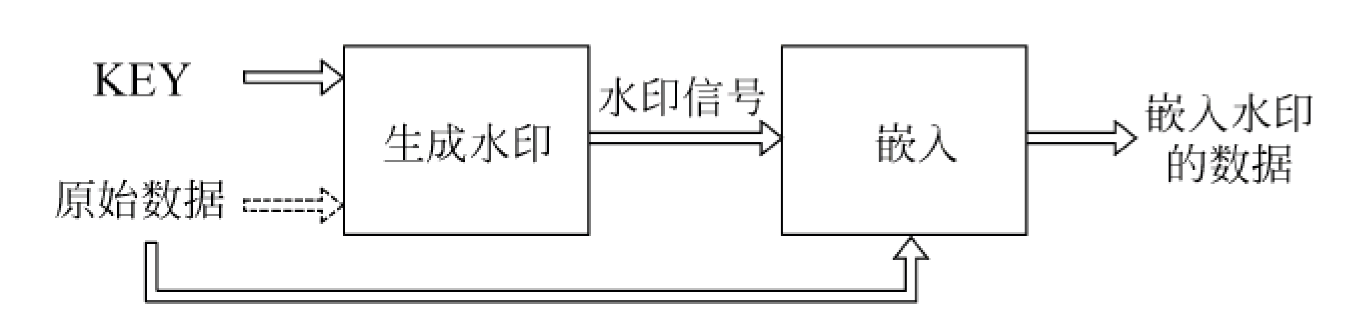


图 5.3.1　常规的水印嵌入模型

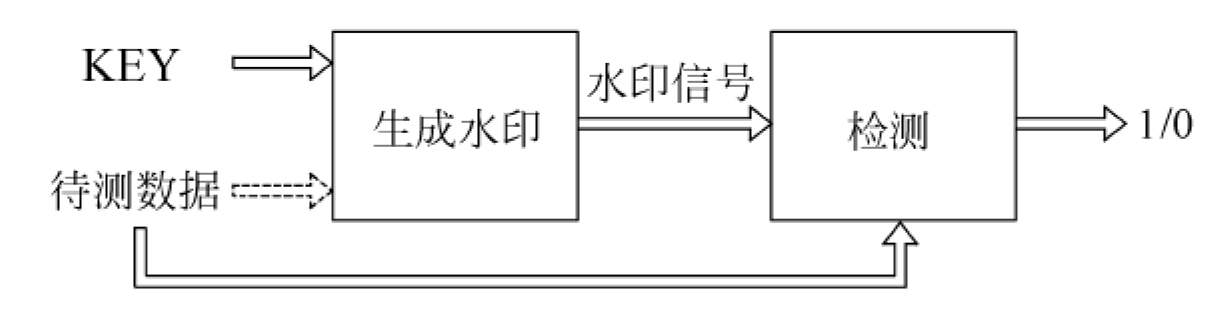


图 5.3.2　水印检测模型

常规的数字水印技术框架可以定义为六元组 (X, K, W, G, E, D)[15]，其中：

（1）X 表示未加入水印的原始数据。

（2）K 表示水印密钥，常由标识数字序列，例如整数序列等组成。

（3）W 表示由数据序列 X 和水印密钥 K 生成的水印信号序列，其定义如下：

$$W = \{w(k) \mid w(k) \in U, k \in \hat{W}^d\} \tag{5.3.1}$$

水印信号可为二值的形式[17]，即水印信号序列中的每个值 $w(k)$，其取值范围 $U = \{0,1\}$ 或 $\{-1,1\}$；也可以高斯噪声的形式[20,24]出现。$\hat{W}^d$ 表示水印信号空间，而 d 表示其维数，$d=1,2,3$ 分别表示音频、图像和视频水印。

(4) G 表示从数据序列 X 和水印密钥 K 生成水印信号 W 的算法：

$$W = G(X, K) \tag{5.3.2}$$

二值形式的水印信号通常基于伪随机数产生器或者混沌系统；而高斯伪噪声信号或者m-序列则可以通过提供很长的不相关信号序列来产生，以保证其足够的安全性。另外，生成的水印可能需要进一步的变换，以便更适合于嵌入到数据中。为了便于分析，可将 G 分解为两个部分：

$$\begin{aligned} &G = T \circ R \\ &\widetilde{W} = R(K), \quad W = T(\widetilde{W}, X) \end{aligned} \tag{5.3.3}$$

第 1 部分中的 R 表示从密钥 K 生成原始水印的变换过程，整个过程仅依赖于 K。如果 R 基于随机数生成器，则 K 可直接映射为生成随机数所需的种子(seed)；而如果基于混沌系统，则 K 可通过一些简单变换以成为混沌系统的初始条件。上述两种情况 R 都满足了密钥的唯一性，并且生成的 $\widetilde{W}$ 是 K 的合法水印；另外，企图通过 R 的逆变换来求得密钥 K 实际上是不可行的。

第 2 部分中的 T 是可选的处理过程，表示将 R 生成的原始水印修改为与被保护数据内容相关的水印，T 只看重数据的一些显著的特征，比如数据在处理过程中比较鲁棒不易丢失的那些特征等。如果在大量数据中嵌入同样的水印，攻击者可以通过统计的方法将数据加以叠加，以估计出水印信号，而通过数据相关的处理，即使采用相同的原始水印，对不同的数据得到的水印也不相同，因此可以避免此类攻击。

(5) E 表示将水印信号 W 嵌入数据序列 X_0 得到加密后的序列 X_w 的算法：

$$X_w = E(X_0, W) = X_0 + af(X_0, W) \tag{5.3.4}$$

其中，最常用且最简单的水印嵌入算法如下：

加法规则：
$$x_w(k) = x_0(k) + aw(k) \tag{5.3.5}$$

乘法规则：
$$x_w(k) = x_0(k) + ax_0(k)w(k) \tag{5.3.6}$$

为了保证在不可见的前提下，尽可能地提高嵌入水印的强度，a 的选择必须考虑图像的性质和视觉系统的特性。

(6) D 表示水印检测算法：

$$D(X, K) = \begin{cases} 1, & W \text{ 存在} \\ 0, & W \text{ 不存在} \end{cases} \tag{5.3.7}$$

不论水印是否受到攻击而造成失真或者水印根本就不存在，都可以通过相似性测量检测出来。有多种办法可以度量原始水印和提取的水印之间的相似程度，最常用的是基于相关性的测试。先用密码和待检测的图像算出水印 W^*，通常情况下提取出的水印 W^* 与原始水印 W 不相等，然后用下面的公式进行计算：

$$\operatorname{sim}(W^*, W) = \frac{W^* \cdot W}{\sqrt{W^* \cdot W^*}} \tag{5.3.8}$$

设定阈值为 T，当满足下面不等式时，W^* 与 W 匹配：

$$\operatorname{sim}(W^*, W) > T \tag{5.3.9}$$

T 的选择要基于一定的虚警概率和漏警概率。检测过程可能包含两个错误，一是实际上没有水印，却检测出有水印；二是实际上有水印，却没有检测到水印。T 减小，则漏警概率降低而虚警概率提高；T 增大，则虚警概率降低而漏警概率提高。

2. 性能评价

在实际应用中，数字水印会面临各种问题，包括数据处理和人为攻击所带来的破坏，大致分类如下：一般信号处理，包括滤波、平滑、增强、有失真压缩等；几何变化，包括旋转、缩放、分割等；诱惑攻击，即试图通过伪造原始图像和原始水印来迷惑版权保护，也称为 IBM 攻击；删除攻击，即针对某些水印方法通过分析水印数据，估计图像中的水印，然后将水印从图像中分离出来并使水印检测失效。水印算法的评估主要有以下 3 种评价标准：

(1) 信噪比 SNR 和峰值信噪比 PSNR

在实验中，我们使用信噪比(signal noise ratio，SNR)和峰值信噪比(peak signal noise ratio，PSNR)作为嵌入水印后图像质量的评估标准，它是一种客观评价标准。信噪比(SNR)和峰值信噪比(PSNR)分别定义为(单位分贝，dB)：

$$\mathrm{SNR} = -10\lg\frac{\sigma^2}{D} \tag{5.3.10}$$

$$\mathrm{PSNR} = -10\lg\frac{M^2}{D} \tag{5.3.11}$$

$$\sigma^2 = \frac{1}{N}\sum_{i=0}^{N-1}(x_i-\bar{x})^2, \quad \bar{x} = \frac{1}{N}\sum_{i=0}^{N-1}x_i \tag{5.3.12}$$

$$D = \frac{1}{N}\sum_{i=0}^{N-1}(x_i-\hat{x}_i)^2 \tag{5.3.13}$$

x_i 表示原图的像素值，$\hat{x}_i$ 表示输出图像的像素值，N 表示图像的像素个数，$[0,M-1]$ 为图像像素值的取值范围。

(2) 水印容量[25]

在给定水印和图像质量标准的前提下，某些水印系统可以测出水印的最大长度和强度。水印容量越大，所含版权信息越多，而不可见性会随之下降。

(3) 鲁棒性

数字水印算法的鲁棒性常用攻击测试来进行评价，常见的攻击测试包括[26]：低通滤波、色彩量化、按比例缩放、剪切、旋转、对称或非对称剪切(X，Y 方向)、对称或非对称行和列移动、普通线形几何变换、JPEG 压缩、小波压缩等。除了上述基本的攻击测试以外，近年来又出现了统计平均攻击和引发多著作权的问题[27,28]。

5.3.3　水印技术分类

5.3.3.1　按照应用媒体分类

数字水印按照应用媒体分类如图 5.3.3 所示。

文本水印，基本上是利用文档的特点，数字水印信息通过轻微调整文档中的某些结构(包括垂直移动行距、水平调整字距、文字特性等)来完成信息嵌入。文本数字水印所采用的算法一般仅适用于文档类，且鲁棒性差，如改变字体之后，嵌入的水印信息就丢失了。

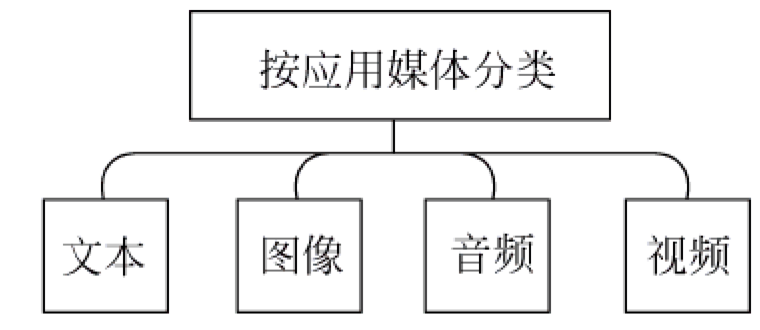


图 5.3.3　数字水印按照应用媒体分类

图像水印，基本上是利用图像的特点，将水印信息嵌入到图像数据中，这种方法利用人类的视觉特点，在

嵌入信息的同时,尽可能地降低对原有图像质量的影响。

音频水印,利用人类听觉系统(human auditory system,HAS)的特点,将水印嵌入到音频信号中,这种水印需要具有很强的鲁棒性和不可感知的特性。

视频水印是图像水印的扩展,它需要实时地提取和检测,与音频水印同样,需要具有不可见性和很强的鲁棒性,以避免压缩编码对水印带来的影响。

5.3.3.2 按照水印特点分类

由图 5.3.4 可以看出,水印技术可以分为可见(visible)水印和不可见(invisible)水印。可见水印用于在媒体中加入明显的版权信息来证明该媒体作品的所有权,主要充当标志和商标等。可见水印的好处是直观,但易于被采用非法手段检测到或删除掉。

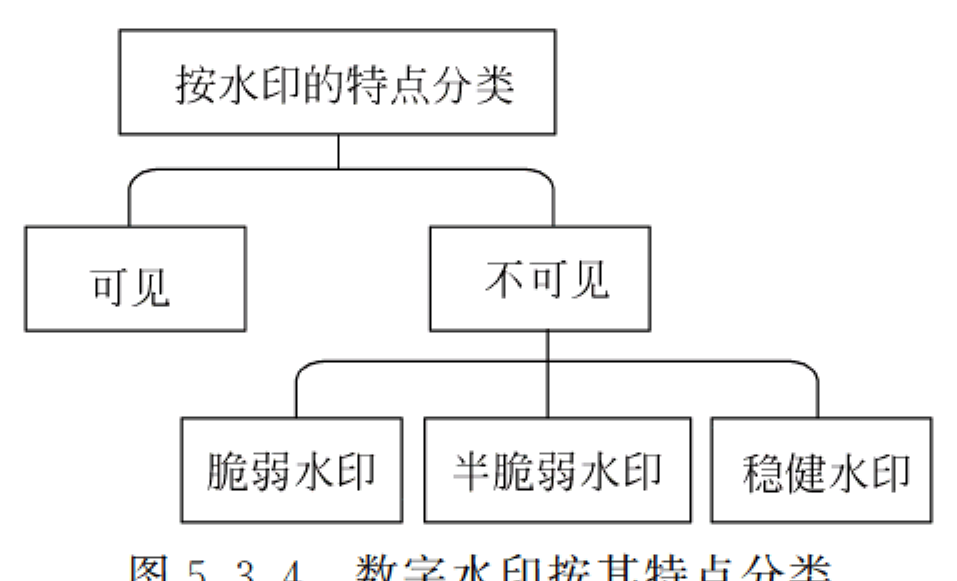


图 5.3.4　数字水印按其特点分类

不可见水印是最常用的水印技术,它利用了人类视觉系统的特点,使得隐藏在数据中的水印无法通过肉眼分辨出来。它可以分为稳健(robust)水印、脆弱(fragile)水印和半脆弱(semi-fragile)水印。稳健水印技术最重要的因素之一就是鲁棒性,即使数据受到较大程度上的修改或损坏,嵌入其中的水印也能保存下来。该技术广泛应用于媒体版权保护、访问控制等领域。

脆弱水印对攻击非常敏感,很容易被破坏掉,这种水印技术通常用来保护信息的完整性。但是在多媒体应用中,加入了认证水印的多媒体内容,很多情况下仍会受到一定程度上的修改,这些修改是在许可范围内的,这时脆弱水印就不再适用了。因此人们采取了折中的办法,提出了半脆弱水印的方法。这种水印技术对一些可接受的保证内容不变的操作(比如压缩和信号增强等)有着很好的鲁棒性,不会受其影响而导致水印破坏;而对于其他恶意的操作(比如增加或删除内容)则很脆弱,很容易被破坏掉。

5.3.3.3 按照水印处理过程分类

图 5.3.5 中按照生成、嵌入和检测过程分别对水印技术进行分类介绍。生成过程中,用作水印信号的可以是噪声和图像。常用于水印信号的噪声包括 3 种:伪噪声序列、高斯随机序列和混沌序列。伪噪声序列是使用最广泛的一类噪声,用它生成水印简单、快速,有很好的自相关函数性质,而且对密码攻击的抵抗性能好。高斯随机序列,即按照高斯分布 $N(0,1)$进行随机生成的序列,该序列同样具有很好的自相关函数性质,常用在多次嵌入水印的提取过程中。混沌现象是非线性动力系统中出现的确定性的、类似随机的过程,它对初值条件敏感,而且有着依赖性,可以提供数量众多、非相关、类似随机而又确定可再生的信号。因此,利用混沌序列作为水印信号,易于生成,而且用它作为嵌入和检测提取信号的密钥不仅简单而且实用。

当用图像作水印信号时,包括二值图案(binary pattern)、邮戳、标志等,其最大的优点在于提取出水印后,可以很直观地辨别出来。并且,用线性反馈移位寄存器、哈希函数等可以使水印生成随机化。混沌序列的缺点是不适用于多次嵌入水印处理。

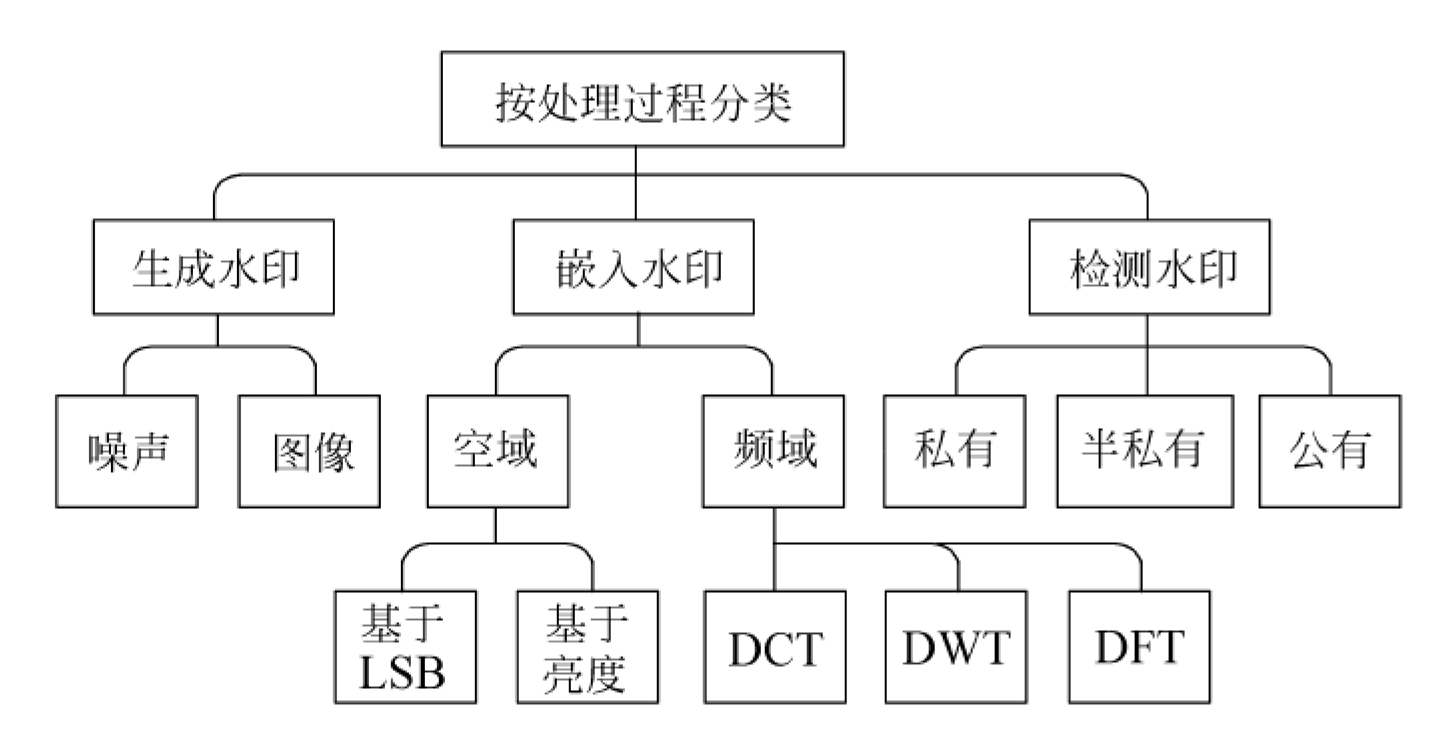


图 5.3.5　数字水印按照处理过程分类

水印按照嵌入过程可以分为空域水印和频域水印两种。其优缺点比较见表 5.3.1。空域水印将数字水印按照某种算法直接叠加到图像的空间域上，称为空域水印算法[29]。因考虑了视觉上的不可见性，水印一般是嵌入到图像中最不重要的像素位(least significant bits，LSB)上，其中最简单的水印方案可以用水印信息代替图像的一个或多个最低有效位平面。另外一种常用方法是利用像素的统计特征将信息嵌入像素的亮度值中，比较常见的有 Patchwork 算法。空域水印实现过程还可以分为基于像素点和基于图像分块两种，LSB 通常是基于像素点的值，而 Patchwork 则是基于图像分块的亮度值。

表 5.3.1　空域水印和频域水印的优缺点比较

分类	优　　点	缺　　点
空域水印	计算速度较快，而且很多算法在提取和验证水印的存在时不需要原始图像 可嵌入的水印容量较大	抵抗图像的几何变形、噪声和图像压缩的能力较差，而且如果确切知道了数字水印隐藏在哪几位 LSB 中，数字水印将很容易被擦除或绕过
频域水印	嵌入的水印信号能量可以分布到空域的所有像素上，保证了不可见性 人类视觉的某些特性(如频率特性)可以很方便地结合到水印编码过程中，有利于提高水印的不可见性 变换域的方法可与国际数据压缩标准兼容，从而实现在压缩域内的水印算法。加快水印嵌入速度	频域水印嵌入和提取信息操作较为复杂，而且嵌入的信息量不能很大

频域水印是将图像作某种变换(一般是正交变换)，然后把水印嵌入到图像的变换域中，称为水印的频域或变换域算法，如 DCT 域、Wavelet 变换域、Fourier 变换域、分形或其他变换域等。通过变换，水印信息将分布到原始数据整个空间里，所以水印一旦嵌入，要将其删除是很困难的。绝大部分算法是基于 DCT 域的，先将图像分割成互不重叠的 8×8 的块，然后对每块进行 DCT 变换，将水印信息叠加到数据信号的中低频分量系数上，然后再 IDCT 变换回去，便得到加入水印后的图像。对不同的水印嵌入算法，都存在有相应的提取和检测算法与之配套，所以在此不对检测算法按前述方法进行分类。

在水印检测过程中，由于媒体不同的需要，水印技术可以分为私有水印、半私有水印和公有水印，它们的区别在于是否需要原始数据。当前大多数的方案都可以归为私有水印和半私有水印之类。私有水印在检测过程中需要原始数据，该技术通过对可能加入过水印的

数据与原始数据的比较来提取水印。半私有水印与私有水印相比，在进行检测的过程中，不需要原始数据。公有水印既不需要原始数据，也不需要嵌入数据的水印信号，可直接从含有水印信息的数据中进行特征统计分析以判断是否嵌有水印，所以也称作盲测。

5.3.4 数字水印典型算法

近年来，国际上数字水印技术的研究发展很快，新技术、新算法层出不穷。水印算法大致可以分为两类，即空域水印和频域水印，后者通常也称为变换域水印，目前很多新的水印算法都是基于变换域的。

5.3.4.1 空域算法

Schyndel 算法[30,31]提出了一些关于水印的重要概念和鲁棒水印检测的通用方法，即相关性检测方法。该算法首先将一个密钥输入到一个 m-序列(maximum-length random sequence)发生器来产生水印信号，然后排列成二维水印信号，按像素点逐一嵌入到原始图像像素值的最低位上。其中，m-序列是由一些初始向量按照 Fibonacci 递归数列的关系运算生成的，也可以用线性移位寄存器实现。如果每个向量的长度为 n，或移位寄存器的级数为 n，则生成的 m-序列长度最大为 2^n-1。m-序列的自相关函数和频谱分布的特点类似于随机高斯噪声。检测的时候，通过计算 m-序列和水印图像行的相关函数来判断是否存在水印。由于 Schyndel 算法将水印信号安排在像素点的最低位上，它是不可见的。但基于同样的原因，水印信息很容易为滤波、图像量化、几何变形等操作所破坏，因此是不鲁棒的。

Patchwork 算法[16]是通过改变图像数据的统计特性来将信息嵌入到像素的亮度值中。Patchwork 算法是随机地选择 N 对像素点(a_i,b_i)，这些随机选取的两个像素点的差值是以 0 为中心的高斯分布。然后将 a_i 点的亮度值加 1，b_i 点的亮度值减 1，这样来改变分布的中心，并且使得整个图像的平均亮度保持不变。最后采用统计的方法来对水印进行检测。为了抵抗诸如有损压缩以及滤波的处理，它将像素点对扩展成小块的像素区域(patch)，增加一个 patch 中的所有像素点的亮度值，同时减少对应另外一个 patch 中所有像素点的亮度值。这种算法对抵御有损压缩编码(JPEG)、剪裁攻击和灰阶校正非常有效。但其缺陷在于嵌入的水印信息少，对仿射变换敏感，对多拷贝联合攻击抵抗力比较弱。

空域水印的其他算法如表 5.3.2 所示。

表 5.3.2 空域水印的其他算法

提出者	算法描述或特点
Delp	在 Schyndel 算法的基础上，将水印序列扩展到二维。具体方法是将图像按照水印图案的大小分为若干个块，然后将水印添加到各块中，将各块拼接起来覆盖整个图像，这样插入的水印序列会更长[18,32]
Cox	将图像分为两组，往其中一组里插入一个正数[20]
Kutter	修改像素点的值，通过与相邻像素点进行比较来检测[33]
Pitas	向图像分块的亮度信息里插入位流数据[34]

5.3.4.2　频域算法

将水印信号嵌入到多媒体信息的部分或所有的频带上，同时可以利用人类的视觉系统(human visual system，HVS)或听觉系统(human auditory system，HAS)特性中的掩盖效应(masking)，在人类感知不敏感的频带加入幅度较强的水印信号，而在人类感知敏感的频带中嵌入较少的水印信息，这样可以在嵌入水印的同时，尽量降低对原始图像或音频信号的质量影响，提高了信息隐藏的可靠性。如果企图消除水印，必须在所有的频带上加入大幅度的噪声，而这将严重损坏数据的质量。版权所有者由于知道水印加入的位置和内容，在验证时很容易把扩散到所有频带上的微弱信号集中起来得到高信噪比的水印信号。

(1) 扩展频谱通信技术

扩展频谱通信(spread spectrum communication)技术的原理为：先计算图像的离散余弦变换(DCT)，然后将水印叠加到 DCT 域中幅值最大的前 K 个系数上(不包括直流分量)，通常为图像的低频分量。若 DCT 系数的前 K 个最大分量表示为 $D=\{d_i\}$，$i=1,2,\cdots,K$，水印是服从高斯分布的随机实数序列 $W=\{w_i\}$，$i=1,2,\cdots,K$，那么水印的嵌入算法为 $d_i^*=d_i+ad_iw_i$，其中常数 a 为尺度因子，控制水印添加的强度。然后用新的系数作反变换得到水印图像 X^*。解码函数则分别计算原始图像 X 和水印图像 X^* 的离散余弦变换，并提取嵌入的水印 W^*，再作相关检验 $\mathrm{sim}(W^*,W)=W^*\cdot W/\sqrt{W^*}$，以确定水印是否存在。该方法即使当水印图像经过一些通用的几何变形和信号处理操作而产生比较明显的变形后仍然能够提取出一个可信赖的水印。

(2) NEC 算法

NEC 算法[20]由 NEC 实验室的 Cox 等人提出，在数字水印算法中占有重要地位。其工作原理是，首先由作者的标识码和图像的哈希值等组成密钥，以该密钥为种子来产生伪随机序列，该序列具有高斯 $N(0,1)$分布。再对图像作 DCT 变换，用该伪随机高斯序列来调制(叠加)图像除直流(DC)分量外的 1000 个最大的 DCT 系数。该算法具有较强的鲁棒性、安全性、透明性等。由于采用特殊的密钥和不可逆的水印生成方法，因此可以有效防止 IBM 攻击。而且该算法还提出了增强水印鲁棒性和抗攻击算法的重要原则，文献[19]建议水印信号应该嵌入到图像频域中可见性最主要的部分，这样可以增强抵抗常规信号处理和几何失真，以提高检测出水印的概率。另外，待嵌入的水印信号要由独立同分布随机实数序列构成，并且该实数序列应该具有高斯分布 $N(0,1)$的特征。

(3) 生理模型算法

人的生理模型包括人类视觉系统 HVS 和人类听觉系统 HAS。利用生理模型的基本思想均是利用从视觉或听觉模型导出的 JND(just noticeable difference)描述来确定在图像或声音的各个部分所能容忍的数字水印信号的最大强度，从而能够避免破坏视觉或者听觉的质量。也就是说，利用生理模型来确定与数据相关的调制掩模，然后再利用其来嵌入水印。这一方法同时具有好的透明性和鲁棒性。

(4) 压缩域算法

基于 JPEG 和 MPEG 标准的压缩域数字水印系统，其水印检测与提取可直接在压缩域数据中进行，节省了完全解码和重新编码过程，因此在数字电视广播及 VOD 中有很大的实用价值[35]。输入的 MPEG-2 数据流可以分为数据头信息、运动向量和 DCT 编码信号块这

3 个部分，常见的方案都主要是对 DCT 编码信号块进行改变，如 H&G 算法[35,36]。具体工作原理是，首先对 DCT 编码数据块中的 Huffman 码进行解码和反量化，以得到当前数据的 DCT 系数块。然后把相应水印信号块的 DCT 系数与视频 DCT 系数相加，从而得到带有水印的 DCT 系数，再重新进行量化和 Huffman 编码。由于 Huffman 编码是变长编码方法，因此加入水印前后的码长可能会发生变化，对于固定码率的 MPEG-2 编码流，需要对新的 Huffman 码字的位数 n_1 与原始数据的码字位数 n_0 进行比较，只有在 $n_1 \leqslant n_0$ 的时候，才传输水印码字，否则传输原码字，这样就保证了不增加视频数据流码率。

另外，由于嵌入的水印信号实际上是一种会降低视频数据质量的误差信号，而基于运动补偿的编码方案会将一个误差扩散和累积起来，可能会影响到帧间编码的 B，P 帧，因此，H&G 算法提出了移位补偿信号。移位补偿信号是加了水印的块和未加水印的块运动补偿预测的差值。在受影响的帧里减去这个差值，就能将误差信号删除。

Langelaar 等人[37]也提出了一种算法，该算法采用了一种新的信息嵌入机制，无需重新编码，通过去掉压缩流的一部分来嵌入标记。除了对 DCT 编码进行修改之外，Jordan 提出的算法[38]还能够将水印信号以一种伪随机方式嵌入到运动向量中，不过该运动向量必须选择指向平坦区域，因为此时修改后，该向量所指的区域不会产生可见的修改痕迹。

（5）其他算法

小波的多分辨率分解算法[29,41,42]是将水印信号和图像进行分解，然后在相应层次上加入水印，最后还原图像，如此可以具有较强的稳健性；基于 DFT 的算法，如算法[19,27]，将水印插入到每个数据块的相位信息里，对压缩和图像处理鲁棒性好；基于分形压缩和编码的水印算法[43]主要是利用分形中的自相似概念和迭代函数系统（IFS），根据拼贴原理在图像的空间域或频域插入水印。

5.3.4.3 主要算法比较

表 5.3.3 对一些常见的数字水印算法的不可见性、鲁棒性、嵌入量以及复杂程度进行了分类比较，以便进一步的研究。总体来说，频域水印的不可见性比空域水印要好，且抗攻击能力很强，但是嵌入量较小，计算更为复杂。实际应用中，需要选择合适的算法，以适应不同的需求。

表 5.3.3 主要算法比较

分类		算法名称	不可见性	抗攻击能力	嵌入量	复杂程度
空域水印	基于 LSB	Schyndel 算法	插入到 LSB 中，不可见性较好	对滤波、图像量化、几何变形等操作抵抗能力很弱	很大	很低
	基于亮度	Patchwork 算法	修改亮度差值分布，不可见性好	对有损压缩编码、剪裁攻击和灰阶校正非常有效。对仿射变换、多拷贝联合攻击比较脆弱	一张图只能嵌入 1 个 bit，水印嵌入量很低	低

续表

分　类		算法名称	不可见性	抗攻击能力	嵌 入 量	复杂程度
频域水印	基于 DCT	扩频通信	不可见性好，但各频段水印强度相同	对几何变形和信号处理操作很鲁棒	嵌入到 DCT 系数上，嵌入量较大	高
		NEC 算法	不可见性好，但各频段水印强度相同	对几何变形和信号处理操作很鲁棒，对 IBM 攻击鲁棒	嵌入到 DCT 系数上，嵌入量较大	比扩频通信更高
		生理模型算法	不可见性好，各频段水印强度不同	对几何变形和信号处理操作很鲁棒	嵌入到 DCT 系数上，嵌入量较大	高
		压缩域算法	不可见性好	对视频压缩、剪辑处理鲁棒性好。某些实现对 QoS 控制机制透明性较差	嵌入到 DCT 系数上，嵌入量较大	较低，避免了 DCT 和 IDCT
	基于 DWT	多分辨率分解算法	不可见性好	对压缩和图像处理鲁棒性好	嵌入到子波段上，嵌入量较大	对于图像分块，高于 DCT 变换
	基于 DFT	算法[19,27]	不可见性好	对压缩和图像处理鲁棒性好	嵌入到每块的相位信息上，嵌入量不大	较高

5.4　视频加密技术

数字水印技术的检测过程通常是通过计算提取出来的水印信息与完整水印之间的相似性，高于一定阈值即可认为水印存在。从这方面来看，数字水印技术不能直接应用于密钥嵌入，因为密钥信息对错误非常敏感，任何一个 bit 的错误都会导致该密钥无法使用。为了保证嵌入密钥的安全性，充分利用了视频图像区域的亮度统计特性，设计了一种基于压缩域水印的视频亮度调制的密钥嵌入算法。

5.4.1　视频加密概述

针对 MPEG 视频的安全传输目前已经提出了很多种加密算法。最直接的方法称为“幼稚”方法，它是将整个 MPEG 流用标准的加密技术来进行加密。但是，由于视频流的尺寸通常很大，所以这种加密方法无法提供很高的处理速度，不适合实时视频加密。

为了提高处理效率，人们提出了选择性加密算法[46,47]。该算法只对 MPEG 视频流中的 I 帧图像进行加密。但是，Agi 和 Gong 等人[48]指出，经过这种方法加密后的图像，仍然能够在局部辨别出图像内容来，这是由于采用帧间压缩编码的帧和一些未加密的帧内压缩块都具有一定的图像校正能力。

Meyer 和 Gadegast[49]设计了一种类似于 MPEG 的视频流格式，称为 SECMPEG。该

格式结合了选择性加密和附加的头信息。但是，该格式不兼容标准的MPEG，而需要有特殊的编解码器的支持。Tang[50]提出了一种整合了压缩和加密在一起的方法。该方法的基本思想是利用一个随机的排列表替换行程编码中的zig-zag顺序。该方法的计算负载很低，但是改变zig-zag顺序会使得视频流的长度增加25%～60%，这样对于视频压缩来说是无法让人接受的。Qiao和Nahrstedt[51,52]开发了一种视频加密算法(video encryption algorithm，VEA)，它将图像的一半用DES/IDEA进行加密，而另一半用"一次填充(one-time-pad)"的方法加密。

Shi和Bhargave[53,54]提出了一种快速的MPEG视频加密算法，该算法通过密钥序列来随机地改变所有DCT系数的符号位以及运动向量的符号位以达到加密的目的。5.4.2节将提出一种简化的实时视频加密算法，该算法基于Shi和Bhargave的方法，二者的区别在于，我们的算法采用一种单向函数(one-way function)来产生加密序列，该加密序列与帧号和会话密钥都相关，这样能够有效地抵抗统计方法的密钥攻击。该算法的处理速度远远超过DES，从而能够满足视频应用中的实时处理的需求。

考虑到视频通信中一定的特殊性，选择性加密算法(selective encryption)的研究成为该领域的研究热点。在多媒体安全领域，选择性加密通常是作为传统的"硬加密算法"(比如AES)的对立面出现的。此类算法并不追求最大的安全性，而是通过降低安全性来换取加密算法的效率，用以满足某个特定多媒体应用的安全需求[55]。在这方面比较经典的算法如下：

(1) 基于DCT变换的视频选择性加密算法

该方法对DCT变换之后的频域系数进行加密，具体的加密方法各有不同。相对于传统的加密整个视频流的方法，有的算法只对视频流中的I帧进行加密，有的则对第n个I帧和由此预测产生的宏块头进行加密，还有的只对视频流中I帧中的亮度或色度分量进行加密。还有一种搜索加密法，是在zig-zag过程中进行加密，采用随机的排列方法将DCT系数块转化为向量。

(2) 基于小波变换的视频选择性加密算法

小波变换的图像编码方法在JPEG2000和MPEG-4中都得到了应用。对基于小波变换的视频的选择性加密只加密那些与变换和编码过程相关的参数，采用与密钥相关的滤波器来重构图像，防止未经许可的正确的重构。

(3) 基于量化树的视频选择性加密算法

量化树(quadtree)是一种图像压缩方法，树的每个节点有0或4个子节点，有0个子节点的节点就是叶节点。如果图像是一致均匀的，则根节点就是叶节点，如果不是，则图像分为4块，同时在树上添加4个子节点。然后对每个新的子节点进行同样的操作搜索。这种加密是对树的结构进行加密。

(4) 面向容错的和视频格式兼容的选择性加密算法

面向容错的方法采用容错加密函数，对每一个传输过程采用不同的排列组合来减少已知的明文攻击，同时减少数据损失。兼容视频格式的选择性加密是对已经采取不同格式压缩后的内容进行加密，从而使加密过程不依赖于格式本身。这种方法的选择性体现在只加密含有内容信息的部分。

综上所述，选择性加密的核心就是以加密整个内容中的重要部分来防止未经授权的观

看，同时保证一定的速度和尽量低的计算量。上述的一些算法虽然各有特点，但由于对视频流语法结构进行了较大的调整，从而降低了处理效率，有的甚至会增大视频流的长度（如改变 zig-zag 顺序）。这些都不能满足视频安全传输的要求，仍然有如下的问题没有考虑到：

（1）应对统计型攻击，由于流媒体数据量较大，这样对统计攻击方法提供了很大的便利性，所以算法一定要考虑如何避免统计型攻击；

（2）容错性，多媒体信息在通信中不可避免地会遇到丢包与误码等问题，加密后的视频流会对丢包更敏感，因为解密时如果有丢包存在的话，可能会使视频质量有较大的下降，如何提高其容错性，对视频的回放质量而言是很重要的；

（3）实时性，对视频流的加解密需要很高的处理效率，以免影响正常的视频回放的效果。

5.4.2　基于应用层组播的密钥管理与分发机制

安全组播中的密钥管理包括密钥生成、密钥分发、密钥更新等内容，根据其实现机制的不同，可以分为如下几类[56]：

（1）基于树的密钥管理算法，即通过逻辑树对组密钥进行管理。在该类算法中，根据密钥树对密钥进行组织和管理，将组播组成员分成多个层次的子组播组，以减小密钥更新消息的长度。典型的算法如：LKH 算法，该算法适用于通常大规模组播中，如 Internet 广播。

（2）基于 Diffie-Hellman 算法的密钥协商管理算法，根据 Diffie-Hellman 算法的基本原理，将用于两方通信的 Diffie-Hellman 算法进行扩展，以支持组播通信，通过组播组内成员共同协商创建、维护组内密钥信息。该类算法主要应用于相对规模较小、具有相对功能较弱服务器（或者无服务器）的组播组，如视频会议。

（3）安全组播框架方案，该方案构建的组播组为分布式系统，系统通过一些可信任的代理管理组内的各个子组，分担主控制服务器的负载，各成员在所属子组内部进行密钥更新，减小了密钥更新消息的长度，主要应用于用户广泛分布的通用组播组。目前，结合应用层组播的特点进行密钥分发和更新机制的研究尚处于起步阶段。不同的应用层组播协议，其覆盖网络的组成方式区别很大，应用层组播的密钥管理要结合组播协议的覆盖网络组成方式进行研究，以满足系统的安全性、可扩展性、实时性和高效性。

密钥管理的目标是要将会话密钥安全地分发给合法用户，以便他们解密组播数据。根据 Rafaeli[57] 的文章，密钥管理方法可以分为 3 种类型：集中式的方法、分布式子组的方法以及分布式的方法。考虑到自适应视频应用的特点，这里着重考虑集中式的方法。

在这些集中式的方法中，我们主要考虑 ELK 方法[58]。ELK 采用了二叉树的数据结构来管理密钥。与其他类似的方案相比，ELK 是一种更加有效、扩展性更好的安全协议，它具有以下特点：

（1）当用户加入时不需要组播密钥更新消息（rekey message），因为此时用户能够自己计算并更新其手中的密钥，而不需要任何额外的资料参与计算；

（2）ELK 中的密钥更新消息长度固定，且比其他方案要短；

（3）ELK 能够可靠地进行密钥更新消息的分发，而不需要可靠组播协议的支持；

（4）ELK 在数据中嵌入了少量的暗示印迹，该方法能让用户通过计算来恢复丢失的密

钥更新消息。总的来说，ELK 在安全性和通信负载上实现了折中。

表 5.4.1 常见的集中式方法的性能比较

		密钥管理的特点				安全性			密钥更新消息的大小		
		基于树	单项函数树	具有可扩展性	密钥恢复	前向机密性	后向机密性	抵抗同谋破解	用户加入		用户退出
									组播	单播	
平面方案	GKMP	否	否	否	否	是	否	是	$2K$	$2K$	—
	Flat Table	否	否	是	否	是	是	否	$2K\log n$	$K(\log n+1)$	$2K\log n$
层次结构方案	LKH	是	否	是	否	是	是	是	$K(2d-1)$	$K(d+1)$	$2Kd$
	OFT	是	是	是	否	是	是	是	$K(d+1)$	$K(d+1)$	$K(d+1)$
	OFCT	是	是	是	否	是	是	是	Kd	$K(d+1)$	$K(d+1)$
	ELK	是	是	是	是	是	是	是	0	$K(d+1)$	n_1+n_2
	a-ary	是	否	是	否	是	是	是	$K(2\log n-1)$	$K(\log n+1)$	$aK\log n$
	a-ary cluster	是	否	是	否	是	是	是	$K(m-1+a\log n/m)$	$K(\log n/m+2)$	$K(m-1+a\log n/m)$

n：组成员个数；a：树的度数；m：cluster 的大小；K：密钥的长度(bit)；d：树的高度；n_1：左子节点贡献的长度；n_2：右子节点贡献的长度。

5.4.3 基于视频的可靠密钥嵌入算法

本节提出的密钥嵌入算法具有较强的差错恢复功能和对一些 QoS 机制(如转码器)的透明性。算法参考了数字水印中的压缩域水印技术，以及许多其他类型的算法，如 Patchwork 算法、NEC 算法等。在压缩域中嵌入水印，可以省去大量的 DCT/IDCT 变换的时间，且有效提高了算法运行的效率。通过修改 DCT 系数以达到调制图像区域平均亮度的效果，从而达到密钥嵌入的目的。该算法在密钥嵌入的基础上，通过将嵌入信息进行 RS 编码和采用信息冗余的办法，因此具有很强的差错恢复的功能和对自适应机制的透明性。另外，算法的设计充分考虑到了实时性和对视频质量的较低的影响，并且能够方便地与已有的密钥分发机制结合起来，为视频组播应用提供访问控制的功能。

5.4.3.1 算法框架

当组播组用户发生变化以后，密钥将会被更新，然后由 GC 分发给各合法用户，分发的过程包含了密钥嵌入。图 5.4.1 描述了用户动态加入或者离开过程中不同的时间段所完成的操作。

首先，欲加入或退出的用户在 $t-2$ 时间段内联系 GC，进行登录或注销的操作。然后，GC 给出更新后的密钥，在 $t-1$ 时间段里将其分发给各合法用户(不包括退出后的用户)。到时间 t 开始的时刻，新密钥生效，密钥更新完成。随着用户不断发生变化，该过程也反复进行。本节中，我们主要研究密钥分发过程中的嵌入算法，在后续的工作中将会和密钥

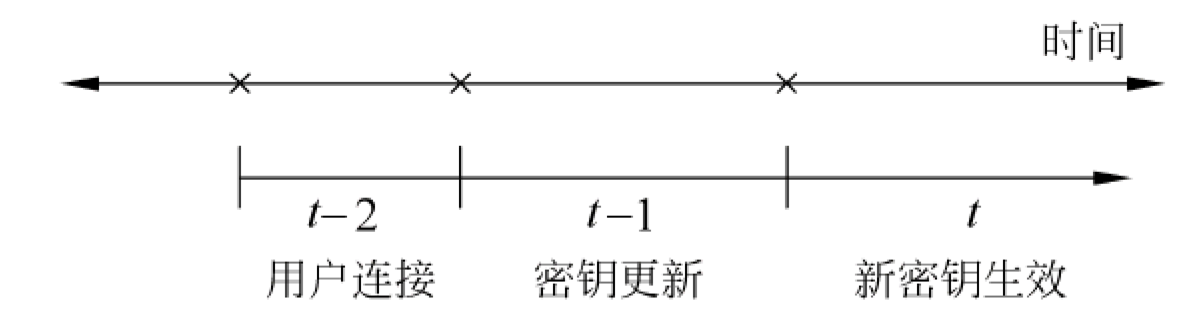


图 5.4.1　密钥更新过程的 3 个时间段

分发管理结合起来。图 5.4.2 给出了该算法的框架(虚线框住部分),即密钥嵌入及检测的过程。

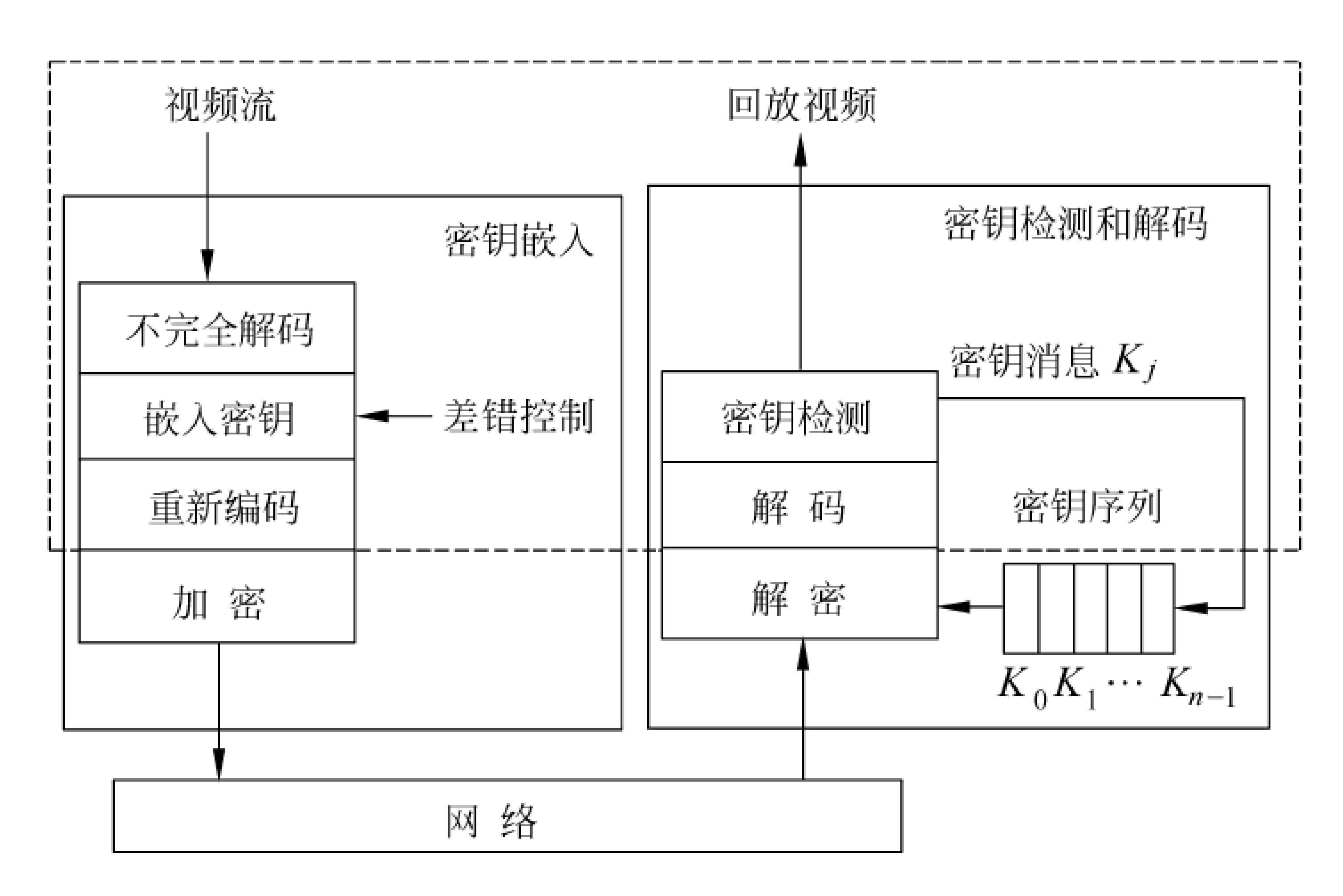


图 5.4.2　密钥嵌入和检测过程

整个处理过程分为两个部分:密钥嵌入部分与解码及检测部分。实际应用中,媒体服务器不可能存放原始视频信息(如 YUV 数据等),那样将造成巨大的磁盘空间浪费,因此,存放的往往都是编码后的视频数据。所以针对这个特点,密钥嵌入部分的输入为编码后的视频流。密钥嵌入只在视频序列的关键帧(即 I 帧)内进行,之所以选择 I 帧,是因为预测帧(P 帧、B 帧)容易被转码器等 QoS 机制中的跳帧算法所跳过,如果嵌入则很容易造成密钥信息的丢失。对于输入的 I 帧,首先经过不完全解码,即变长解码(variable length decoding, VLD)和逆量化(inverse quantization, IQ),从而得到 DCT 系数。算法通过修改每个 8×8 DCT 系数块的直流分量从而达到密钥嵌入的目的。因为直流分量与该 8×8 像素块的平均亮度直接相关,对其进行修改就能达到调制像素块平均亮度的目的,而不必通过 IDCT 变换得到原始数据。另外,嵌入的密钥信息采用了 Reed-Solomon 纠错码,以提供差错恢复的能力。然后,将修改后的 DCT 系数通过量化、变长编码(variable length encoding, VLE)后得到视频流,完成整个嵌入工作。当视频流在有差错的网络中进行传输时,往往会有丢包的情况发生,因此该算法采用了信息冗余的方法,即对每一个嵌入了密钥的图像序列组(group of pictures, GOP,包含 1 个 I 帧和多个 B 帧、P 帧),接下来的 1 个或者多个 GOP 中均嵌入了相同的密钥信息,一旦前面 GOP 中的密钥信息丢失,可以通过后续的 GOP 来恢复,这样增加了密钥的可靠性。图 5.4.3 给出了一个例子,利用 4 个 GOP 来携带新密钥。

视频流经过加密后才能进行传输,加密采用的密钥为旧的会话密钥,以达到访问控制的目的。解码和检测通常在接收端进行,接收端首先通过旧密钥对视频流进行解密,然后对其

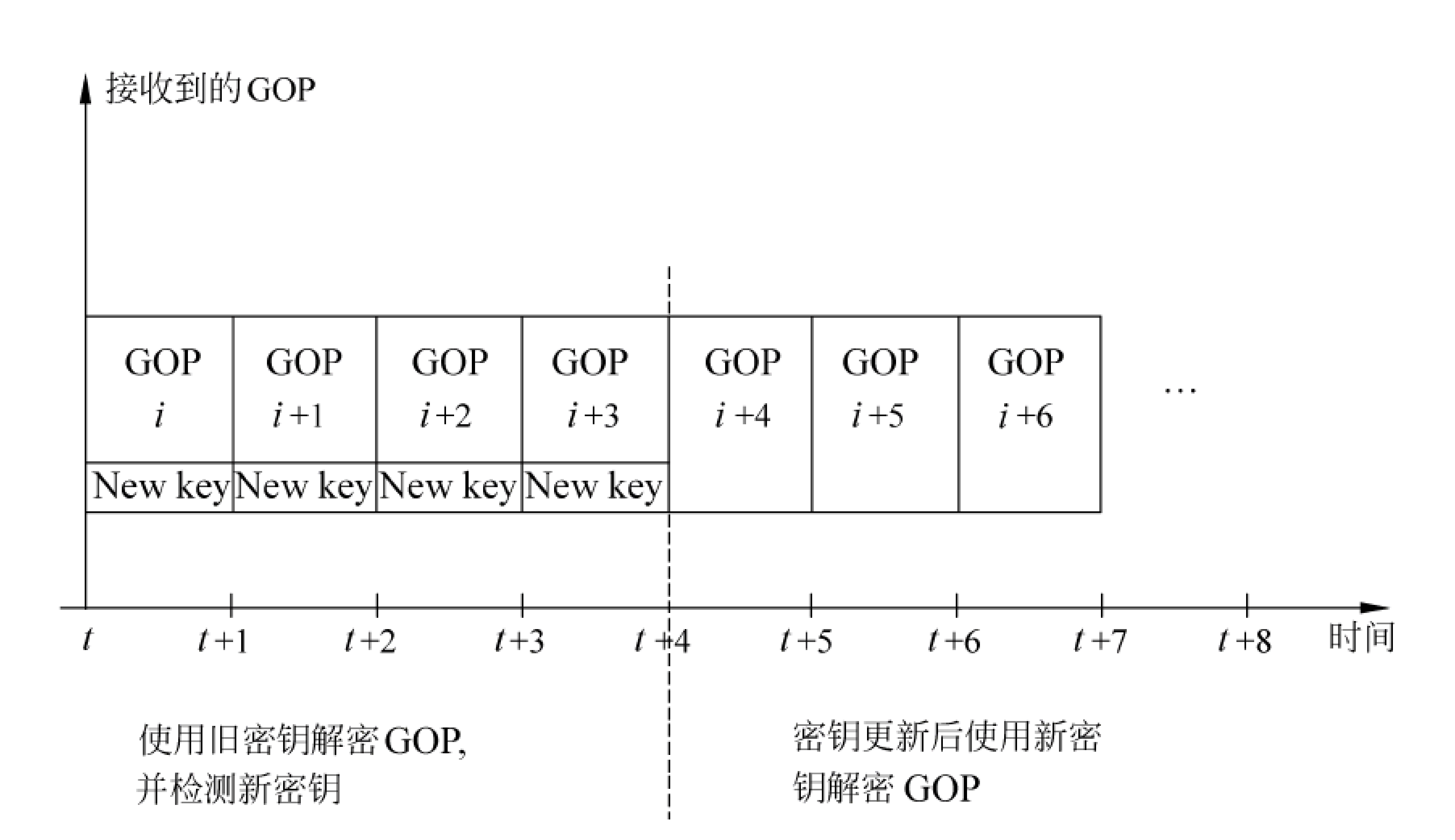


图 5.4.3 密钥分发过程中的密钥冗余

解码并回放。同时检测是否有密钥更新的信息,如果有,则获取该更新后的密钥。待密钥分发过程完成,发送端和接收端同时启用新的密钥,完成密钥更新。

5.4.3.2 算法原理

1. DCT 系数与图像亮度的关系

在 MPEG/JPEG 图像压缩标准中,图像各点亮度值被划分为若干 8×8 的块,对每个块进行二维 DCT 变换得到对应的 8×8 DCT 系数块。该系数块坐标最低的位置,即(0,0)位置存放的是直流分量,向横纵坐标方向移动,存放的是高频分量。由 DCT 变换的公式:

$$F(u,v)=\frac{2}{N}c(u)c(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)y\pi}{2N}\right] \quad (5.4.1)$$

对 DCT 系数块直流分量有

$$F(0,0)=\frac{1}{N}\sum\sum f(x,y)=N\left(\frac{\sum_{x=0}^{7}\sum_{y=0}^{7}f(x,y)}{N^2}\right) \quad (5.4.2)$$

取 $N=8$,可得:

$$F(0,0)=8\left(\frac{\sum_{x=0}^{7}\sum_{y=0}^{7}f(x,y)}{64}\right)$$

故由式(5.4.2)可知,DCT 系数块直流分量大小为该 8×8 亮度块平均亮度值的 8 倍。如果该块内所有像素点的亮度值都增加(或减少)Δ,则其对应的 DCT 系数块的直流分量增加(或减少)8Δ。另外,如果保持交流分量不变,直流分量增加(或减少)Δ,则所有点的亮度值增加(或者减少)Δ/8。根据这个关系,我们可以直接通过修改 DCT 系数块直流分量的值来修改对应区域的平均亮度值,从而节省了计算量。

2. 亮度调制

我们利用图像的亮度统计特性,对一个像素区域的平均亮度值(average luminance

value，ALV）进行调制，以嵌入密钥信息。一个区域由若干个 8×8 的像素块组成，每个区域嵌入 1 个 bit 的信息。图 5.4.4 为调制后的亮度分布。

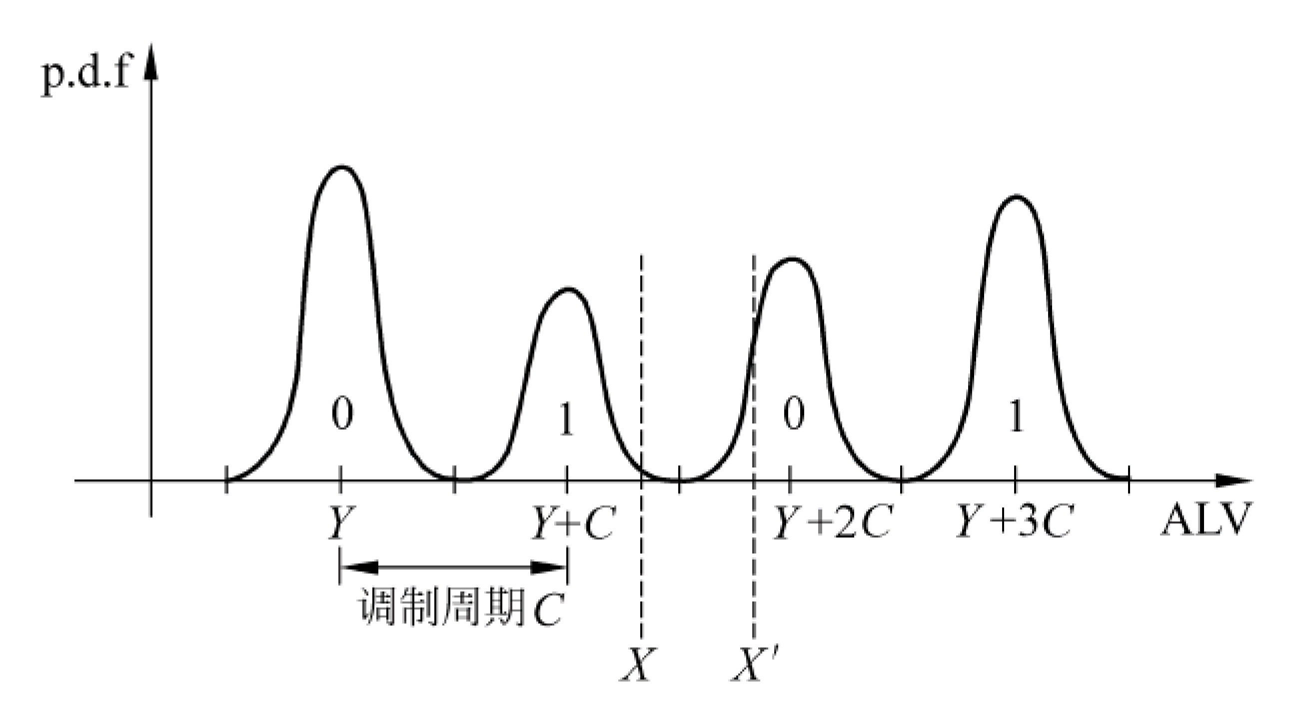


图 5.4.4　调制后的亮度分布

如图 5.4.4 所示，横坐标为图像区域的平均亮度值，其中 Y 为[0,255]范围内的一个固定值，由发送端和接收端共享，用于防止亮度值超出[0,255]的范围。C 为空域中亮度值的调制周期，实验中其取值范围为 2～16，对应 DCT 域中的调制范围为 16～128。图中一系列值为 $Y+nC$（n 为非负整数）的参考点代表了嵌入的 1 个 bit 数据，当 n 为偶数时，表示嵌入的 bit 为 0；当 n 为奇数时，表示嵌入的 bit 为 1。调制的过程是根据要嵌入的 bit，将原始的平均亮度值调整至最近的 $Y+nC$ 上。例如，图中所示某个区域的平均亮度值为 X，拟在该区域嵌入 1，则需要将其移动至 $Y+C$ 处。移动时，首先计算移动的距离，即 $Y+C$ 与 X 的差值，然后将该区域内每个 8×8 块的平均亮度值都加上这个差值，从而完成移动。

检测时根据平均亮度值所在的区间来确定嵌入的 bit 是 0 或者是 1，如果该值在某个 $Y+nC$ 的$[-C/2,C/2]$的范围内，则按照这个 n 的奇偶性来确定嵌入的值。例如，图中平均亮度值 X'，其所在区间为$[Y+2C-C/2,Y+2C+C/2]$，故可知嵌入的 bit 为 0。前面提到，嵌入过程将会把图像区域的平均亮度值移至一系列的 $Y+nC$ 上，但是传输过程中受转码器等的影响，会产生一定程度的偏差。因此，通过判断平均亮度值所在区间来确定嵌入的数据，能够极大地减小这种偏差对检测带来的影响。可见，调制周期 C 的选择决定了检测区间的大小，C 越大，则区间越大，检测正确性越高，然而因为 C 也是亮度调制的最大距离，所以对图像质量的影响会越严重。C 的选择需要考虑转码器造成的误差，下面将对此进行详细的阐述。

3. 视频转码器的影响

视频流常常需要在异构网络和计算能力不同的设备之间进行传输，不同的带宽和图像处理能力决定了各接收端对视频质量的需求。例如，宽带用户能够接受到高码率、高质量的视频，而 PDA 由于是无线传输加之本身处理能力较弱，所以需要接收低码率、低分辨率的图像。为了满足各种用户的需求，如果单纯在服务器端存放同一视频片段的各种质量的版本，那么必将会造成存储空间爆炸。因此，需要在各接入端对视频进行转码，使之变为满足该网络或者用户的需求，这种机制被称为视频转码器。转码器可以实现视频流码率整形、视频流格式转换、空间分辨率调整等功能。

5.4.3.3 亮度调制算法

当视频流进入到密钥嵌入模块时，首先会进行变长解码(VLD)、逆量化(IQ)，从而得到若干 DCT 系数块。对图像区域的平均亮度的调制，实际上就是对各 DCT 系数块的直流分量进行修改，从而使得其平均值改变为某一指定值(对应于空域亮度值 $Y+nC$)。然后重新进行量化和变长编码，以得到嵌入密钥后的视频流。图 5.4.5 给出了该密钥嵌入过程，其中 C_D 为原始的 DCT 系数的直流分量，C_D^M 为调制以后的直流分量。

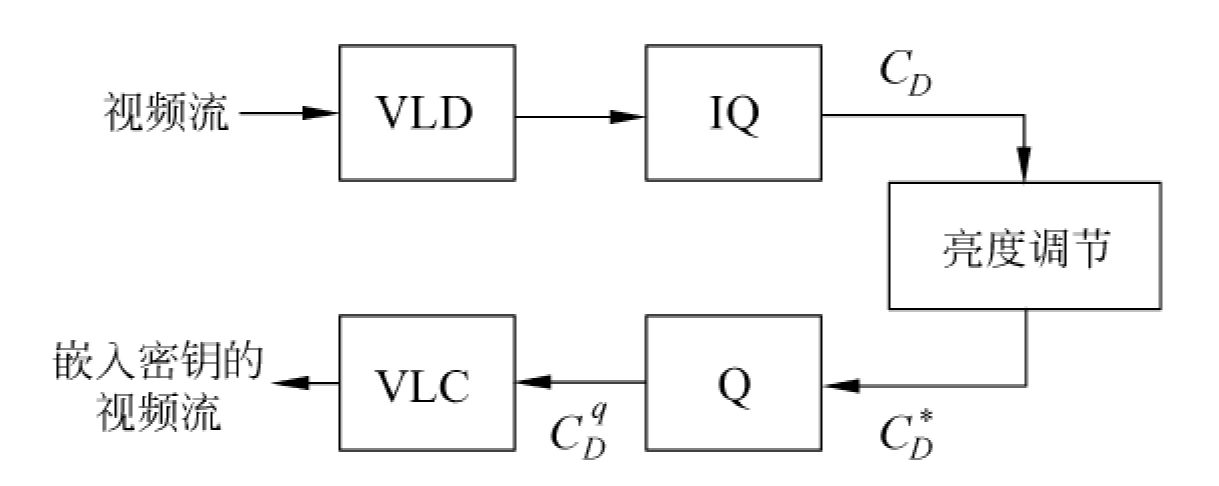


图 5.4.5 密钥嵌入过程中的亮度调制

假设某个区域由 N 个 8×8 块组成，各块平均亮度值为 ALV_i，该区域的平均亮度值为

$$\mathrm{ALV}=\frac{\sum_{i=1}^{N}\mathrm{ALV}_i}{N} \tag{5.4.3}$$

如果为了嵌入某个数据需要使 ALV 增加 $\Delta(|\Delta|\leqslant C)$，理论上可以通过将每个 ALV_i 都增加 Δ 来实现，即：

$$\mathrm{ALV}+\Delta=\frac{\sum_{i=1}^{N}(\mathrm{ALV}_i+\Delta)}{N} \tag{5.4.4}$$

但是，由于量化的关系，使得每个块增加的值实际上并不等于 Δ，导致 ALV 无法增加 Δ。为了使 ALV 准确地移动至指定值，我们提出了一种亮度调制算法，通过将前次亮度修改中的理论增加值与实际增加值的差值累计起来，到下一次修改时一起增加。算法流程如图 5.4.6 所示。

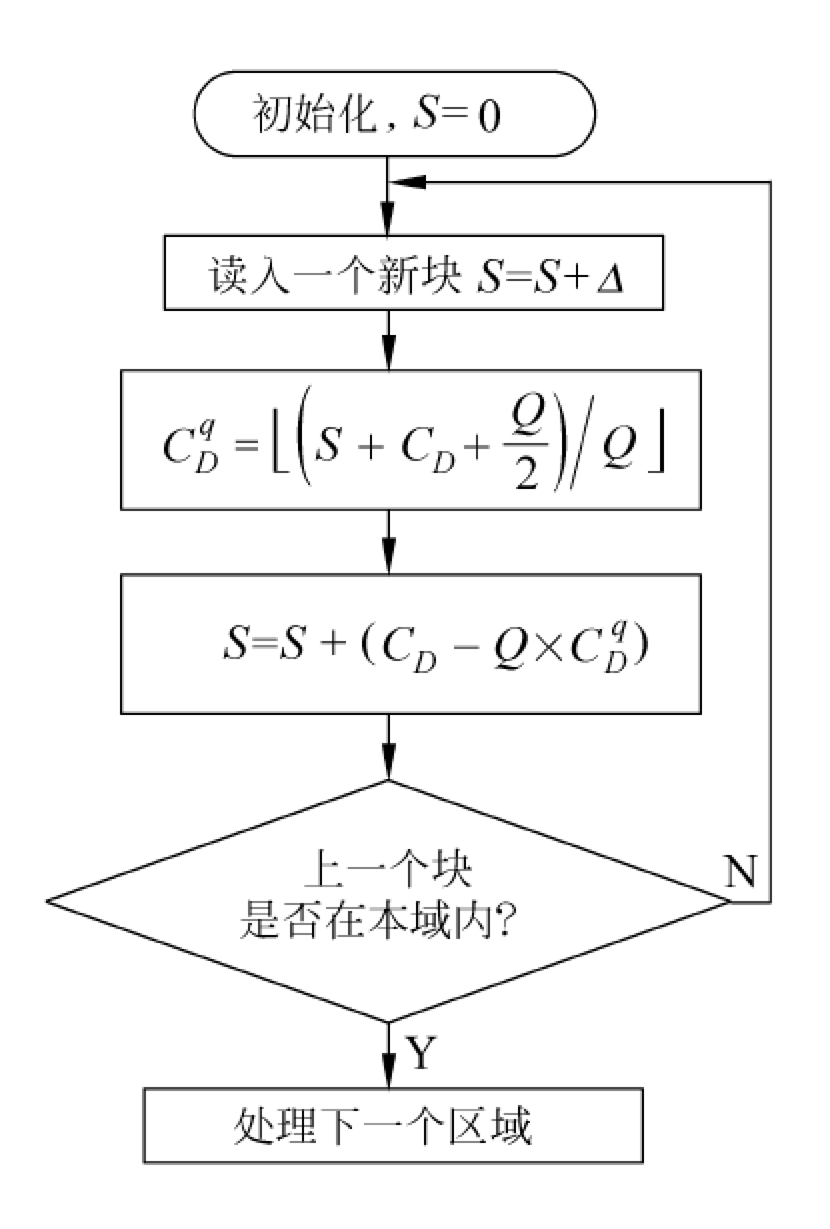


图 5.4.6 亮度调制算法流程图

图 5.4.6 中，S 为累加器，负责累计对每个块计算后产生的误差。例如在 DCT 域中，某个块的直流分量值 $C_D=20$，量化器大小为 $Q=10$，需要增加的 $\Delta=4$。由式(5.4.3)、式(5.4.4)计算可知，就算 C_D 被修改为 24，经过重新量化后，C_D 仍然等于 20。此时，产生误差值为 4。下一个块的 $C'_D=10$，需要增加两个量：$\Delta=4$ 和前次误差 4，即为 18。经量化后实际变成 $C'_D=20$，又产生误差值为 -2，累计到下一次的计算中，直到该区域内最后一个块。求其平均值，则该区域亮度变化误差绝对值不大于 Q。换句话说，嵌入过程调制得到的平均亮度

值与其期望值(某个 $Y+nC$)之差 $|\Delta_{\text{Embed}}|\leqslant Q$。当 N 足够大时，该误差为 0。此时，检测过程的准确性将只受转码器的影响。

5.4.3.4　图像区域划分

前面提到，每一个图像区域存放 1 个 bit 的信息。图像区域的划分，需要兼顾嵌入信息量和可靠性的要求。对于前者，考虑到常见密钥长度为 128 位，加上一些标志位和校验位一共 200 位，故将图像划分成 200 个区域。若区域太多，会使得每个区域内的块数过少，对抗转码器等 QoS 机制的能力减弱。从实验结果可以看出，分为 200 个区域是不错的选择。

考虑到嵌入密钥的可靠性，尤其是经过下采样后块数减少，如果组成区域的 8×8 块过于分散，嵌入其中的密钥信息将会因为所在块被丢弃或者与相邻块作平均而被丢失或破坏。对于前面提到的常见的空域下采样方法，即通过取平均值的办法将图像长宽各减少一半，原图像中的每个宏块(16×16)变为 1 个 8×8 的像素块，二者的平均亮度值相同。基于这些考虑，图像区域的划分应尽量以宏块为基本单位，并使得各宏块之间聚集在一起，这样能够保证下采样使得分辨率下降以后，新的 8×8 块能与原区域对应，以保证密钥的可靠性。

图 5.4.7 所示是对大小为 640×480 的图像的一种最简单的划分方法。该方法是对图像按比例进行分区，接收端也只需按相同比例在变小后的图像寻找相应的区域。其优点是实现简单，且对多种比例的图像分辨率变化都比较可靠。但是由于区域中的块都集中在一起，各块的亮度性质较为相似，如果转码过程中产生误差，各块也很相似，这样不利于减小误差。另外，嵌入密钥后可能会使图像出现比较明显的一些亮块，对图像质量有一定的影响。改进的办法是可以将区域中的块 4 个一组分散到图像中，但是这样划分的方法比较复杂，而且可靠性不如前者强。因此，这里采用图 5.4.7 所示的分区方法。

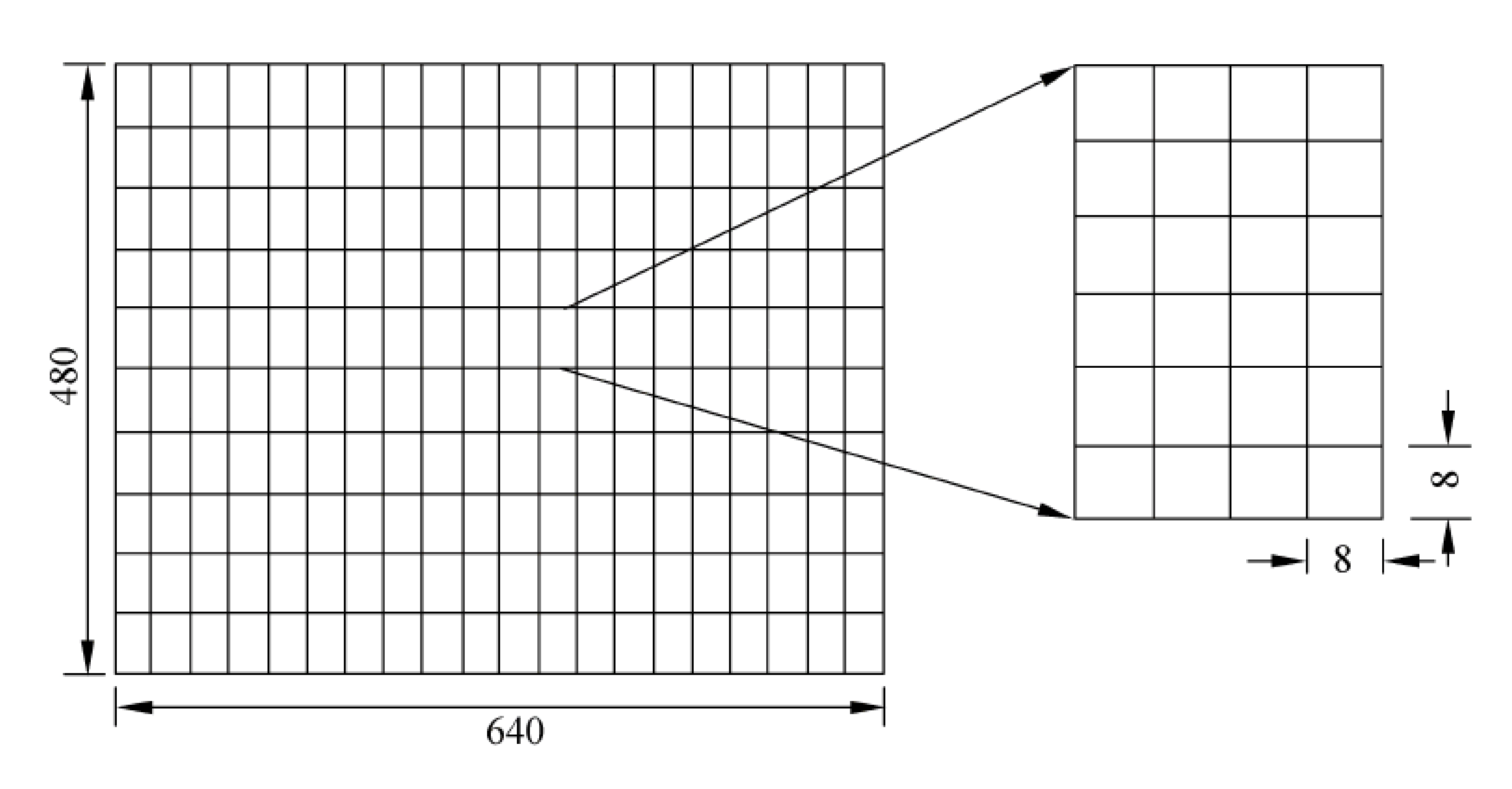


图 5.4.7　640×480 图像的一种区域划分的方法

5.4.3.5　差错恢复

1. Reed-Solomon 编码

Reed-Solomon(RS)码[59]是一种性能优良的分组线性码，在同样编码冗余度下 RS 码具

有很强的前向纠错(forward error correction,FEC)能力。同时,由于近年来超大规模集成电路(VLSI)技术的发展,使原来非常复杂、难以实现的译码电路集成化,目前功能很强的、长 RS 码的编译码器芯片也商业化了,因此 RS 码已广泛应用于通信领域。

RS 码是一类非二进制 BCH 码,在(n,k)RS 码中,输入信号分成 km 比特一组,每组包括 k 个符号,每个符号由 m 比特组成,而不是 BCH 码中的一个比特。一个纠 t 个符号错误的 RS 码有如下参数:

码长:　　$n=2^m-1$ 个符号,　　或者 $m(2^m-1)$比特

信息段:　　k 个符号,　　或者 km 比特

监督段:　　$n-k=2t$ 个符号,　　或者 $m(n-k)$比特

最小码距:　　$d=2t+1$ 个符号,　　或者 $m(2t+1)$比特

具有 $2t$ 个符号的监督段的 RS 码,能够纠正 t 个错误的符号。它至少能够纠正 t 个比特的分散在各符号上面的错误,至多纠正 t 个完全错误的符号,即 tm 个比特的连续错误。所以,RS 码特别适合于纠正突发错误。

为了提高嵌入密钥的可靠性,增强其对转码器的抵抗能力和对较低丢包率情况下的错误恢复能力,我们采用了RS码对可能出现的少量错误进行纠正。考虑到嵌入的信息为 200 位,其中 128 位为密钥信息,剩余 72 位中取 64 位用作 RS 码的码字,8 位用于存放一些属性。所以采用(25,17) RS 码,8 个字节的监督段,可以纠正 4 个字节上的错误,至少 4 位至多 32 位。图 5.4.8 所示是这 200 位信息的组成。

8 位标识
128 位密钥信息
64 位奇偶校验码

图 5.4.8　嵌入的 200 位信息的组成

2. 信息冗余

利用 RS 码可以极大地减少因为计算误差和转码器共同引起的检测错误。但是,视频流在进行网络传输的过程中,会受到网络环境的影响。网络突发的拥塞会导致大量分组的丢失,而通常一个关键帧的比特流会被分成十多个包(如 UDP,RTP 等)进行传输,遇到突发的丢包情况就会造成嵌入信息的大量错误,使得 RS 码无法对其进行纠错。此时,该受损的密钥信息便无法使用,而将被丢弃。考虑到视频对延时的敏感性和视频组播的特点,不可能采用传统的重传机制,所以可以考虑采用信息冗余的方法来对其进行恢复。假设第 i 个关键帧第一次携带了新的密钥信息,将其后若干个关键帧$(i+1,i+2,\cdots,i+k)$均嵌入相同的密钥信息,这些冗余帧的个数由服务器端通过一定的策略来决定。当第 i 帧丢失或者受到严重损坏时,可以通过 $i+1,i+2,\cdots,i+k$ 帧中正确接收的一帧来对密钥信息进行恢复。一旦接收到了正确的密钥,便忽略后来的关键帧所带的冗余信息。

因此,关键帧被分成了 3 种类型:更新帧、冗余帧和普通帧。前两者是携带了更新后的密钥信息的关键帧,而普通帧则不带密钥信息。为了避免普通帧中含有某些相同的信息而被误认为是更新帧,所以也要对其进行"密钥"嵌入,以表明其类型。所有这些可以通过在嵌入密钥信息的 8 位标志位中加以标注。图 5.4.9 给出了这 8 位标志位的取值和含义。

通常情况下,每两个关键帧之间平均间隔约 0.5 秒,所以一般来说 64 个关键帧对于一次密钥更新来说完全足够了。如图 5.4.9 所示,最高两位为帧类型标志,00 为普通帧、01 为

7	6	5	4	3	2	1	0

位	数值	含义
7~6	00	普通帧
	01	更新帧
	10	冗余帧
5~0	63~0	剩余的冗余帧数量

图 5.4.9　标志位的取值和含义

更新帧、10 为冗余帧。低 6 位为此冗余帧后面的冗余帧的数量，当该值为 0 时，从下一个关键帧开始用新的密钥进行解密。

采用冗余帧相当于为嵌入的密钥加上了双保险，能够非常有效地解决密钥嵌入的差错恢复的问题。这两种方法，实际上是通过修改嵌入信息来实现的，或者说是在密钥嵌入算法的上一层进行的操作。因而具有很好的独立性，可以对这些方法分别进行研究，以提供更强的差错恢复功能和对特殊情况的适应性。

5.4.3.6　实验结果与分析

模拟实验的环境如图 5.4.10 所示。对于原始信号，我们采用了两段 MPEG-2 测试序列，一段来自于电影《侏罗纪公园》，另一段是现场抓取的视频，大小均为 640×480，速率为 20 fps，都是 500 帧。选择这两个序列是根据它们不同的运动和场景特征，前者含有快速的动作和场景切换，而后者运动相对较小，而且场景基本不换，这样便于对密钥嵌入算法进行较全面的测试。

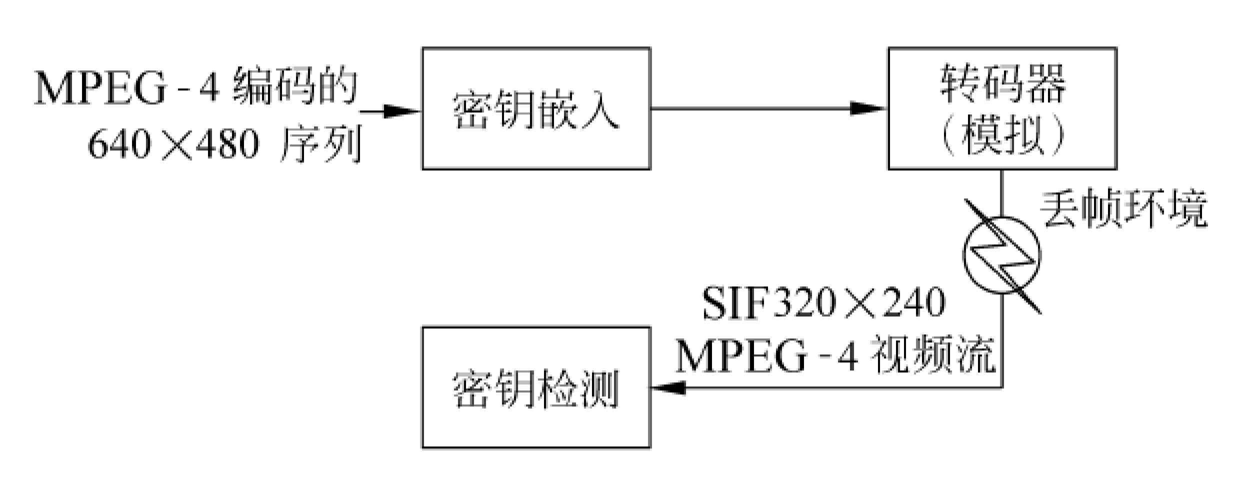


图 5.4.10　实验框架

为了测试方便，并没有采用实际使用的转码器，而主要模拟了转码器中空域下采样和重新量化这两个重要的操作。另外，还加入了丢帧的模块，以模拟在网络传输过程中突发的丢帧情况。

图 5.4.11 所示是各种取 C 值情况下，对图像质量的影响。除了第 1 张图片外，剩下的 3 张图分别用不同的调制周期进行密钥嵌入。从中可以看出，当 C 的取值小于 4 时，密钥嵌入所造成的影响可以忽略不计。而当 C 大到一定程度时(如图 5.4.11(d)所示)，图像中出现了很多或亮或暗的块，图像质量下降得很严重。

从图 5.4.12 中可以看出，当调制周期 C 取值不超过 4 时，PSNR 的下降基本没有超过 0.5 dB，这对于视频质量来说是很理想的。而当 C 取到 16 时，很明显地，PSNR 下降超过了 3 dB，而且其 PSNR 波动得非常剧烈，此时图像质量下降很多，无法令用户接受。



(a) 未嵌入



(b) 调制周期为 2



(c) 调制周期为 4



(d) 调制周期为 16

图 5.4.11 嵌入密钥后的片段在不同调制周期下的效果

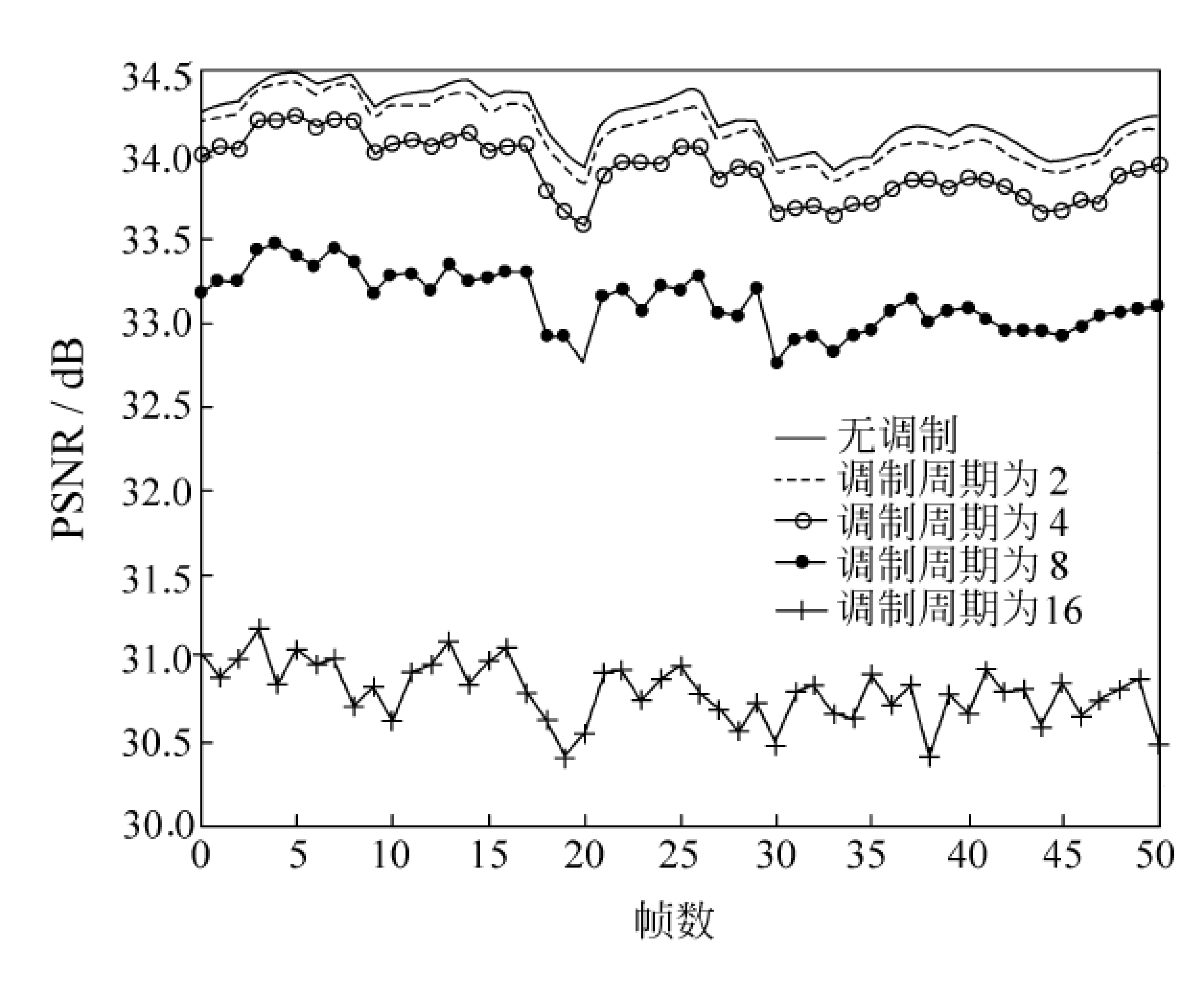


图 5.4.12 图像峰值信噪比(PSNR)与调制周期的关系

图 5.4.13 所示,是通过转码器后检测出的平均误码位数与调制周期的关系。用不同大小的原量化器量化得出的亮度值,经过转码器中新量化器以后,产生的误差各不相同,图中的 3 条线分别代表 3 种原量化器和新量化器相配合时产生的平均误差位数(总数为 200 位)。可以看出,当调制周期小于 3.5 时,即对应于 DCT 域的调制范围 28,检测出错的概率很高,如图中所示最高约为 15/200×100%=7.5%的出错率。而当调制周期快到 4 的时候,错误率已经趋近于 0。

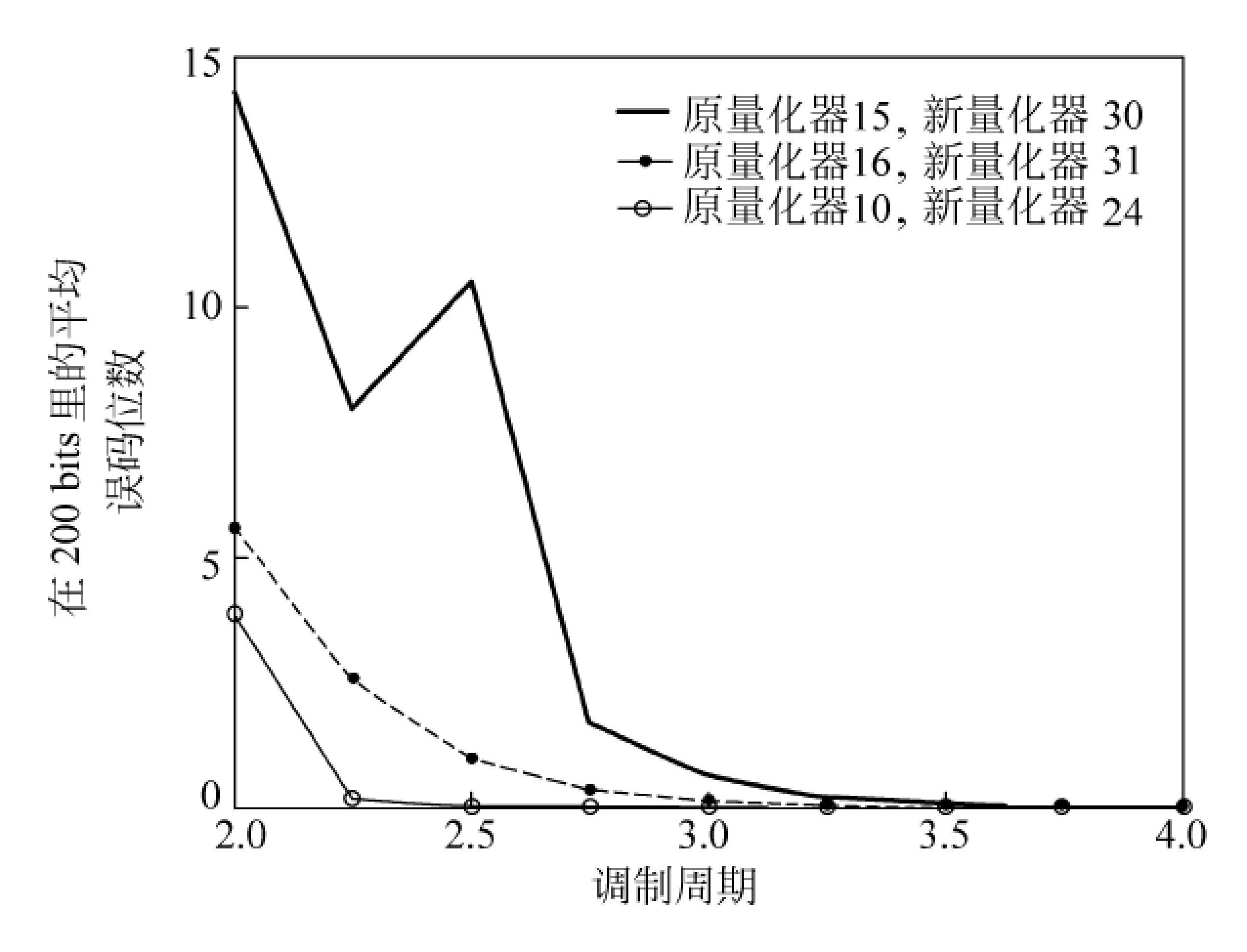


图 5.4.13　通过转码器后检测出的平均误码位数与调制周期的关系

因此，采用通常情况下的量化器进行重新量化，调制周期取到 4，已经足够使得检测错误概率大为降低，并趋于 0。由于忽略了下采样过程中的计算误差，在一些极端的情况下(如下采样、重新量化都造成了最大的误差，且符号相同)，仍会出现一些错误。但是，这样出错的概率非常低，可以通过 RS 码对其进行纠正。

图 5.4.14 给出了丢包率与检测出错概率的关系，并将同样大小的关键帧分别打成不同数量的包来进行发送。从图中可以看出，出错率随着丢包率的增加呈阶梯性的递增，这是因为密钥信息的检测是对整个区域进行的，即使部分块受到损坏，其余占多数的正确块也能保证该区域的正确性。另外，同样大小的关键帧如果采用过多的包来传输，虽然每个包携带密钥信息的数量减少了，但在相同的丢包率下，丢失的包变得更为分散，影响到的图像区域也更多，因此最后的差错率会比一般情况高得多。如果采用了 RS 码，检测出错的概率发生了很明显的下降。比较图 5.4.14(a)和(b)可以看出，在使用了(25，17)RS 码的情况下，丢包率小于 10%时，出错率已经降至 0 附近。而对于丢包率较高的情况，出错率下降了一半以上。可见，RS 码能够对嵌入密钥的可靠性起到很大的作用。

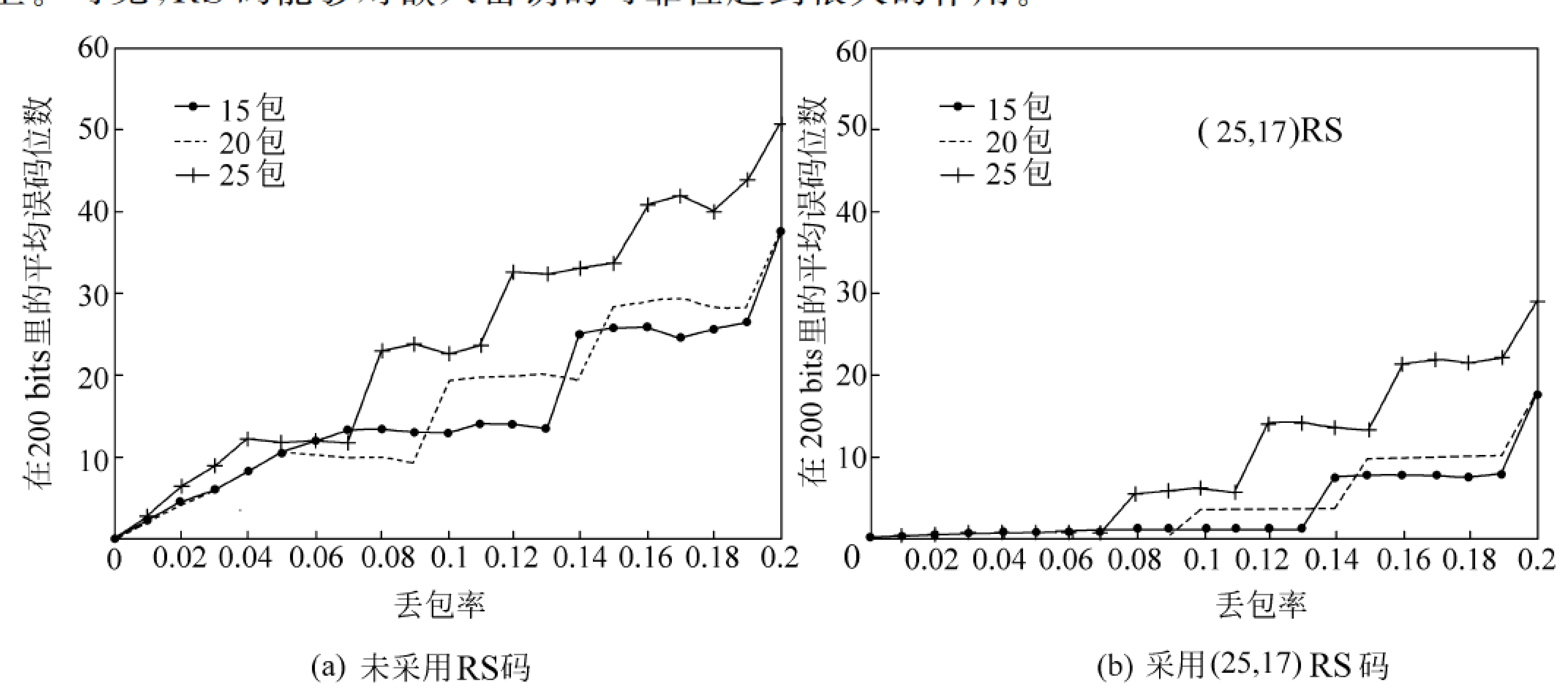


图 5.4.14　丢包率与检测出错概率的关系

在更高丢失率的情况下，即便采用 RS 码也无法避免错误的发生。此时，我们采用冗余帧来进行差错恢复。考虑到网络中的丢包情况常常是突发的，即一段时间内有严重的丢包，之后又恢复良好的网络状况。故只要更新时间允许，可以使用较多的冗余帧，使得接收端总可以获得正确的密钥信息，从而正确地完成密钥的更新过程。通常情况下，一个由 15 帧图像组成的 GOP 持续时间大概在 0.5 s。由于密钥更新过程不会频繁地发生，接下来的很多 GOP 就可以用作前面 GOP 中嵌入的信息的冗余备份，因而密钥分发的时间也就延长到了 $0.5(n+1)$s（n 为冗余 GOP 的个数）。在这段时间内，解码端能够检测到若干份的密钥信息的复制，然后选择其中正确的来更新它自己的密钥。

考虑到视频的实时性，密钥嵌入操作不应占用过多的时间。因此，我们通过将这两个片段进行重新编码，并且对比正常的编码时间和编码后再进行密钥嵌入所需全部时间，来对算法效率进行说明。通过设定编码器，使得连续两个关键帧之间不会超过 10 个预测帧。测试平台 CPU 为 AMD Athlon 1 G，内存 256 MB。结果见表 5.4.2。

表 5.4.2　密钥嵌入算法带来的对处理时间的影响

序 列 名 称	《侏罗纪公园》片段	现场录像
正常编码速度/(f/s)	52.20	53.41
含有密钥嵌入的编码速度/(f/s)	47.07	49.17
增加的处理时间的百分比/%	11.4	9.28

从表中可以看出，密钥嵌入仅增加了 10%左右的计算时间，可见不会对视频传输带来太多额外负担。通常间隔若干帧才会出现一个关键帧，因此对这一帧进行处理所增加的时间是微不足道的。

5.4.4　基于视频的选择性加密算法

本节提出了一种高效的视频选择性加密算法，该算法采用伪随机函数来生成加密序列，并且采用两层加密的方法，分别对 I 帧 DCT 系数块的 DC 分量的符号位和其余帧的运动向量进行修改以实现加密的功能，因此具有很高的安全性。另外，由于该算法只对视频内容在 DCT 域中的数据进行加密，因此对服务质量控制机制具有很好的透明性，并且具有实时处理的能力，另外不会增加视频码流的大小。

5.4.4.1　加密序列

这里提出的视频选择性加密算法采用了两层的方法，分别对 I 帧 DCT 系数块的 DC 分量的符号位和其余帧的运动向量进行修改以实现加密的功能。修改过程是将它们与一个伪随机序列按位做异或运算，该序列称为加密序列(encryption sequence，ES)，能够由会话密钥 K_G 和帧的编号通过 PRF 函数的运算而得到：

$$\mathrm{ES} = \mathrm{PRF}_{K_G}^{(n\to n)}(\mathrm{frame_number}) \tag{5.4.5}$$

该序列随着帧号而发生变化，这样能够抵抗很多统计攻击。由于用户知道会话密钥和帧号，他们能够自己计算出加密序列来，从而能够对视频进行解密。

PRF(pseudo random function)是一种密钥相关的伪随机函数，通常是一个密钥相关的哈希函数，根据一定的输入得到确定的输出。PRF 用来派生密钥和认证处理。在后边的部分，我们用到了很多伪随机函数，这些函数具有不同的输入和输出长度。为了表述方便，我们定义了一组 PRF，都是将 m 位的输入 M 与密钥 K 经过计算得到 n 位的输出，其定义如下：

$$\mathrm{PRF}_K^{(m\to n)}: K\times\{0,1\}^m \to \{0,1\}^n \tag{5.4.6}$$

5.4.4.2　第一层加密

对于 I 帧，我们选择了像素块的亮度来作为加密的对象。根据上文的介绍，DCT 系数块的 DC 分量与一个块的平均亮度值直接相关。另外，在 MPEG 视频压缩标准中，DC 分量是按照预测的方式进行编码的。改变一个 DC 符号位码字将会对后续块重建的 DC 分量造成严重的影响。具体的改变方法是，将 DC 分量的符号位与加密序列中对应的位进行异或运算。图 5.4.15 给出了 DC 系数加密的示意图。

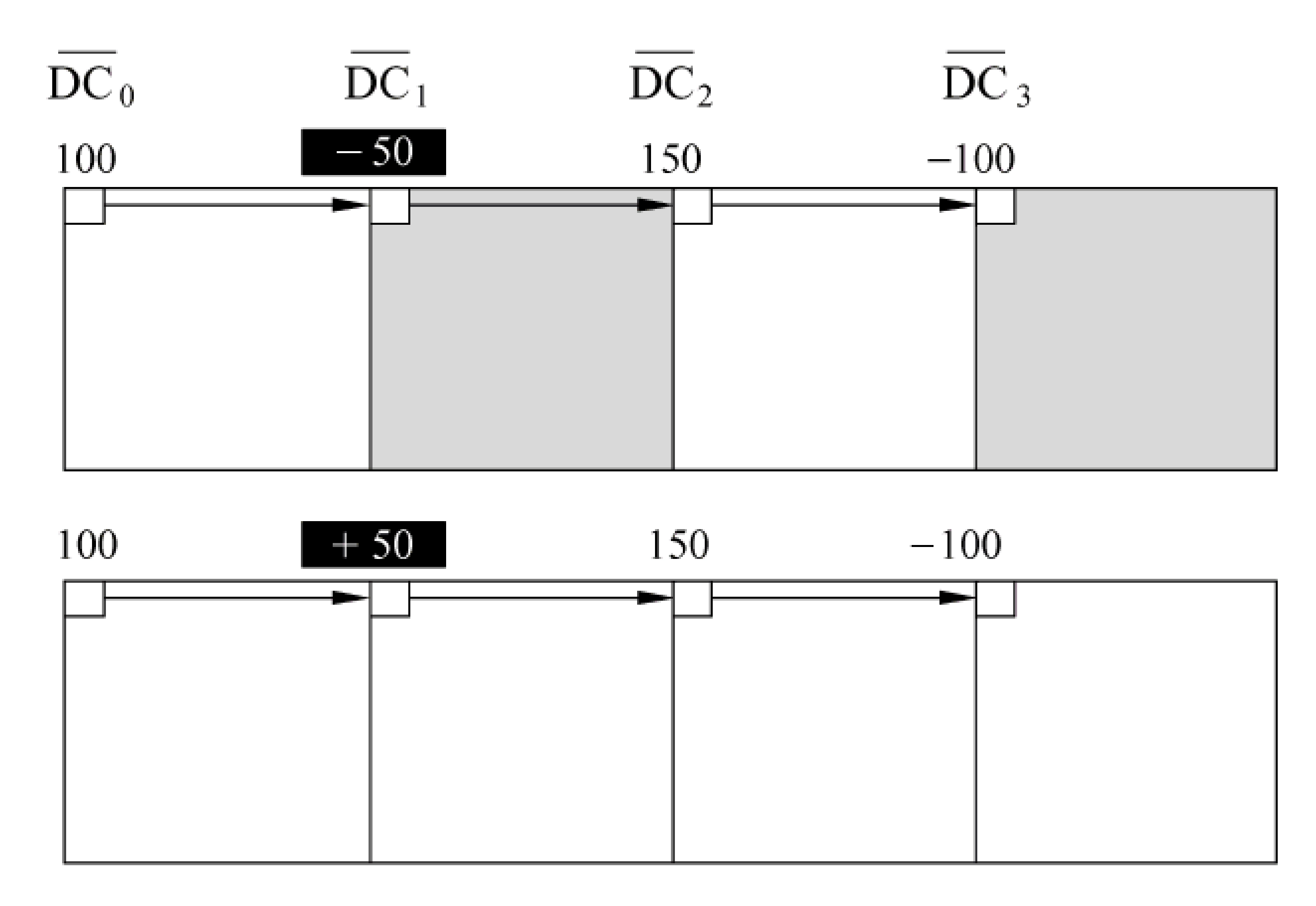


图 5.4.15　第一层加密的示意图

假设$\overline{\mathrm{DC}_i}$是待传输的码字，DC_i 是对应的真正的 DC 分量。我们有：

$$\overline{\mathrm{DC}_0}=100,\quad \mathrm{DC}_i=\overline{\mathrm{DC}_i}+L \tag{5.4.7}$$

设 K_i 是 ES 的第 i 位。我们可以通过下面的计算来进行加密：

$$\mathrm{Enc}(\overline{\mathrm{DC}_i})=(2\times(\mathrm{sign}(\overline{\mathrm{DC}_i})\oplus K_i)-1)\cdot|\overline{\mathrm{DC}_i}| \tag{5.4.8}$$

其中，$\oplus$表示异或操作。$\mathrm{sign}(\overline{\mathrm{DC}_i})=\begin{cases}0，若\overline{\mathrm{DC}_i}<0，\\1，若\overline{\mathrm{DC}_i}\geqslant 0。\end{cases}$

举例来说，如果$\overline{\mathrm{DC}_i}>0$ 且 $K_i=1$，那么加密后的码字为$-\overline{\mathrm{DC}_i}$。如图 5.4.15 所示，改变任意一个 DCT 系数块的 DC 符号位，将会造成后面整个图像的亮度混乱。这将极大地降低图像的可辨识性，从而对 I 帧图像达到很好的加密效果。

5.4.4.3　第二层加密

对于帧间编码的帧(如 P 帧、B 帧等)来说，运动向量足够用来保存一些低分辨率视频对象的信息，这样将使得图像中某些对象能够被辨识出来。另外，这些帧具有对 I 帧的修复功

能，即使 I 帧内容被加密，随着后续 P、B 帧的解码，重建后的图像越来越倾向于恢复到未加密的样子，这将导致第一层加密的实效和视频信息的泄漏。考虑到运动向量的数据相比亮度、色度信息来说具有更高的压缩比和更少的数据量，因此将其作为第二层加密的对象是不错的选择。

假设当前宏块的运动向量是 MV，而 MV1，MV2，MV3 是相邻宏块的运动向量。我们可以得到：

$$\begin{cases} Px = \text{Mid}(\text{MV}1x, \text{MV}2x, \text{MV}3x) \\ Py = \text{Mid}(\text{MV}1y, \text{MV}2y, \text{MV}3y) \end{cases} \tag{5.4.9}$$

其中，Mid()表示取输入的中间数。例如，我们可以得到：Mid(2,3,5) = 3。因此，差分码字 MVD 为

$$\begin{cases} \text{MVD}x = \text{MV}x - Px \\ \text{MVD}y = \text{MV}y - Py \end{cases} \tag{5.4.10}$$

设 K 为 ES 的某一部分，故可以通过下面的计算来实现加密：

$$\text{Enc}(\text{MVD}) = \text{MVD} \oplus K \tag{5.4.11}$$

与 DC 分量的加密方法类似，运动向量也是经过预测编码的。少量的运动向量的修改将会造成图像重建时大量的宏块位置混乱。图 5.4.16 给出了采用第二层加密对图像带来的影响。

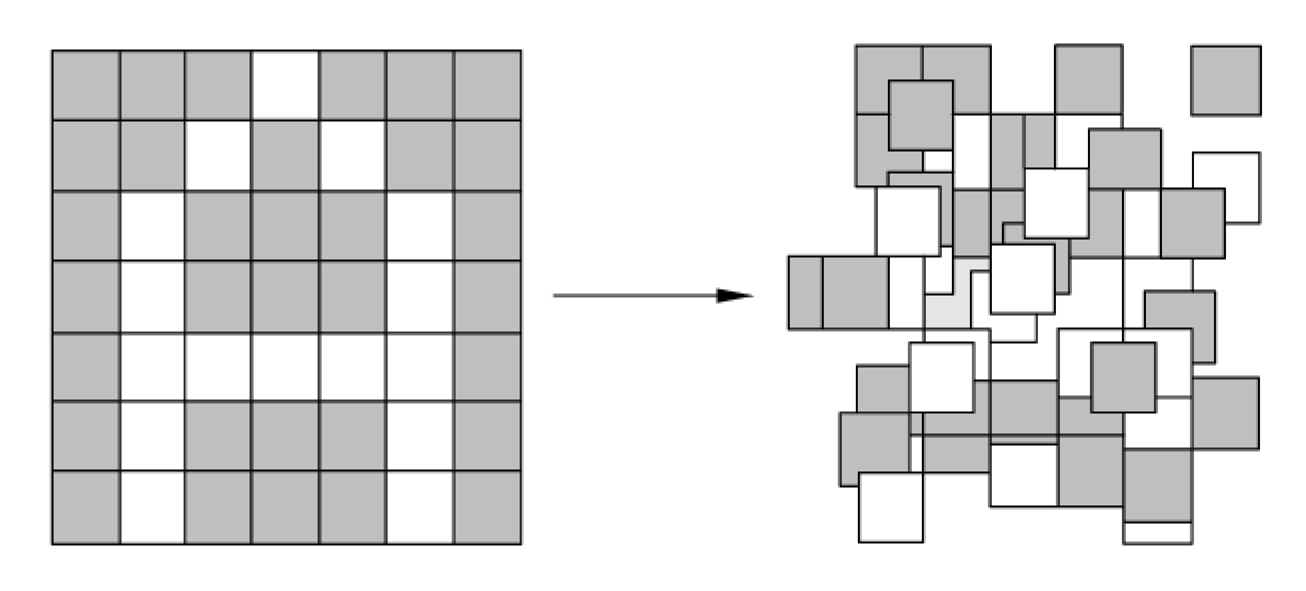


图 5.4.16　第二层加密的示意图

5.4.4.4 实验结果与分析

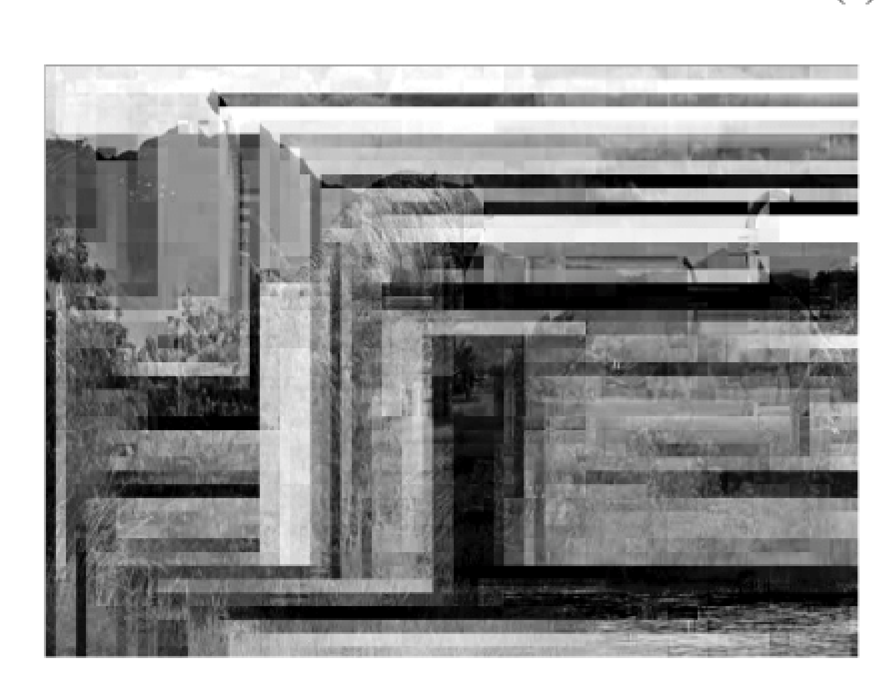
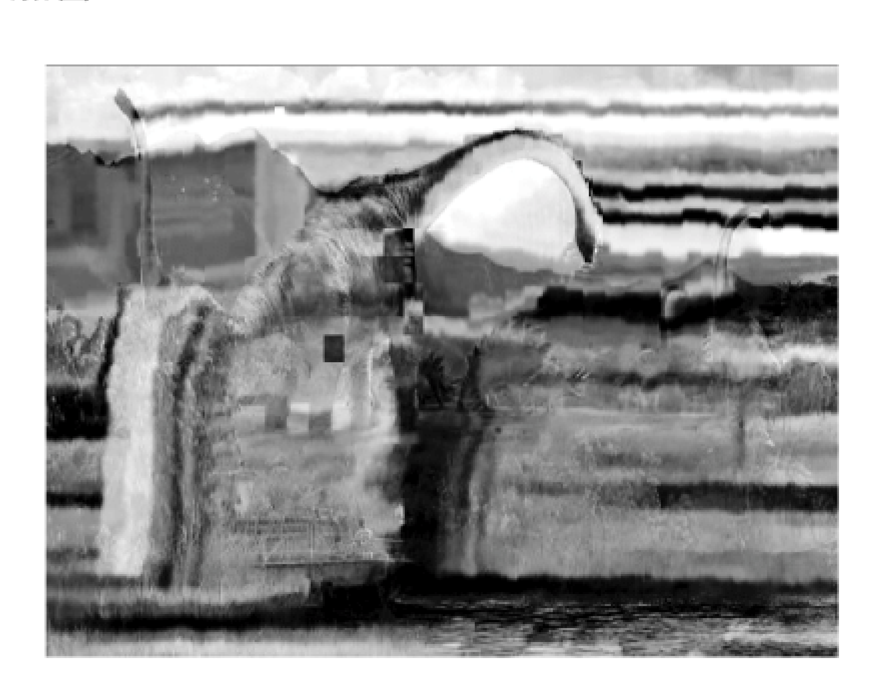
如图 5.4.17(b)所示，在经过第一层加密的视频图像中，有很多亮度或深或浅的条纹，其图像质量受到了很大影响，基本无法辨识其中的内容。然后随着后续的 P 帧的播放，右边图像中出现了恐龙的轮廓。如果经过两层加密，情况就不同了。如图 5.4.17(c)右图所示，图像内容已经完全无法辨别出来。在前面的描述中可以看到，我们的选择性加密算法只进行了一些简单运算，因此它具有很低的运算复杂度，表 5.4.3 也给出了它在编码过程中增加的处理时间。

表 5.4.3　选择性加密算法的处理时间

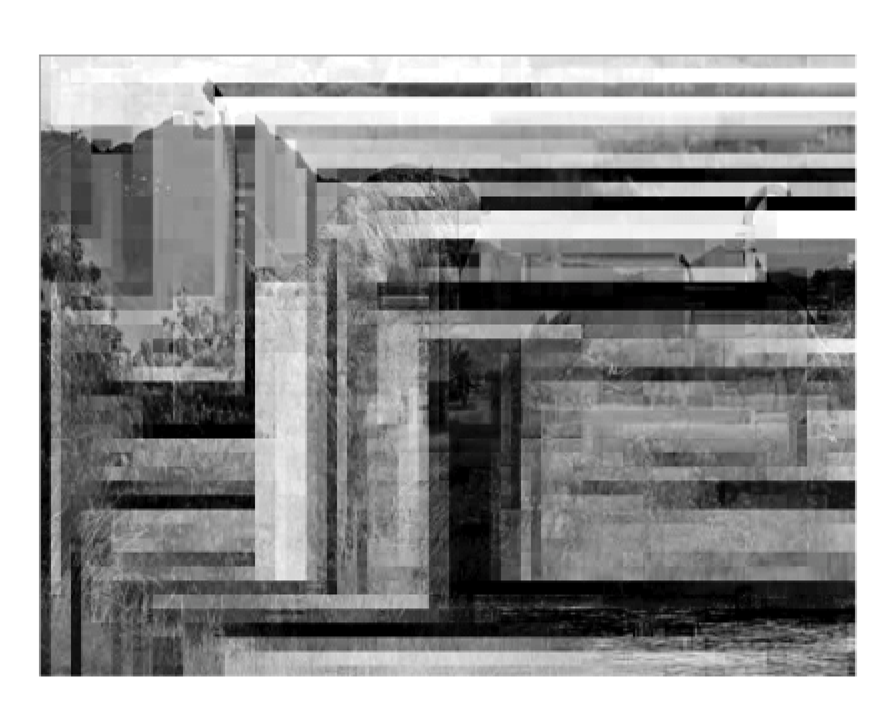
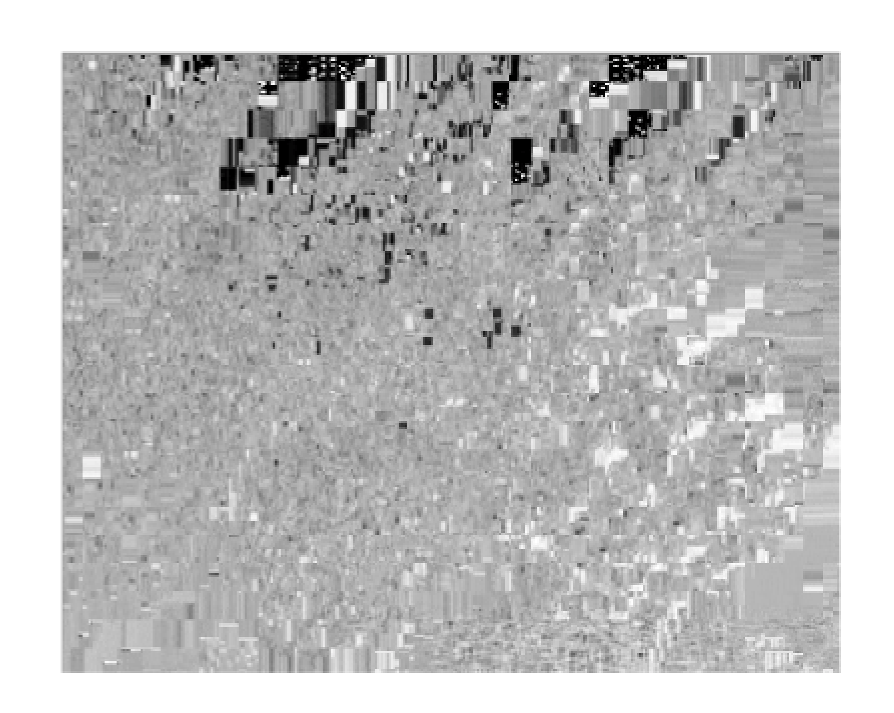
编码后的图像质量	高	低
编码速度，不采用加密算法/(f/s)	49.7	55.6
编码速度，采用加密算法/(f/s)	48.4	54.3




(a) 未加密

(b) 第一层加密

(c) 两层加密

图 5.4.17　加密效果(左列图像来自于 I 帧,右列图像来自其后的 P 帧)

增加的处理时间/%	2.69	2.39

另外,我们还选取 Foreman 序列进行了加密测试,该序列为 CIF 图,帧率为 30 fps,总共有 300 帧。其加密效果如图 5.4.18 所示。从图中可以看到,画面中的 Foreman 已经完全无法分辨出来。表 5.4.4 给出了该序列进行选择性加密所增加的计算负载,从表中可见,选择性加密算法仅仅增加了不到 3%的编码负载和不到 1%的解码负载。

表 5.4.4　选择性加密算法的计算复杂度

	无加解密操作	有加解密操作
编码速度/(f/s)	50.4	49.1
解码速度/(f/s)	112.8	112.2



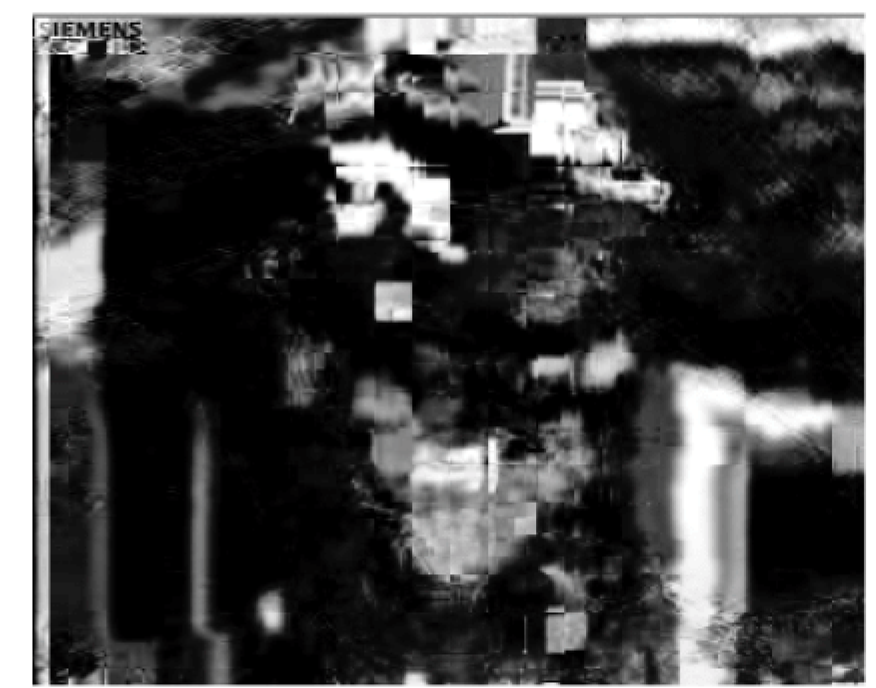


图 5.4.18 Foreman 加密效果

增加的负载/%	2.64	0.54

5.5 媒体相关的视频安全组播协议——MSMP

媒体相关的密钥分发方式具有性能和安全方面的双重优势。上节提出了媒体相关的密钥分发方案中所需的两种关键算法,本节将综合以上两种算法,并采用 LELK 算法作为密钥管理与分发算法,针对自适应的应用层视频组播提出一种新的媒体相关的安全组播协议(media-dependent secure multicast protocol,MSMP),该协议将为处于开放和不安全的网络环境中的自适应视频应用提供安全支持。该协议在现有的组播协议基础上加入若干关键部分,包括可靠的数据嵌入技术、实时视频加密算法以及具有差错恢复和高扩展性的密钥管理方案。

MSMP 综合了 3 个主要的部分: 密钥管理 LELK 算法、可靠的密钥嵌入算法以及视频选择性加密算法。MSMP 的主要优点在于: 它对自适应机制具有透明性,即使下层组播协议无法提供可靠的传输服务,它也具有很好的鲁棒性;它不增加带宽的需求,也不会造成存储空间的激增,即使是在组播组规模很大的时候;能够提供实时处理的能力。实验结果表明,MSMP 能够为视频传输提供安全保障,以抵抗攻击和自适应机制带来的损害。

5.5.1 MSMP 框架

MSMP 方案具有以下特色: ①它能够提供很高的安全性,从而减轻了交换机和路由器上相关的需求; ②构建于现存的 IP 组播基础上,因此能够很容易地进行部署; ③对自适应机制和差错信道具有很好的鲁棒性; ④具有很高的扩展性,能够很容易地对大规模的组播组进行部署。

MSMP 由 3 部分组成: 密钥管理、密钥嵌入和视频安全传输,每部分都由单独的层次来进行处理,如图 5.5.1 所示,分别是: 会话层、密钥分发层和安全传输层。当用户加入或者退出的时候,密钥必须更新以保证后向和前向的安全性。如图 5.5.2 所示,会话层的密钥管理实际上是对一系列密钥按照树型的结构进行操作。每个用户对应于一个叶节点,拥有并

维护其所在的从叶节点到根节点的一条路径。当用户事件发生的时候，服务器更新密钥树上相应的密钥，然后通知用户更新他们的密钥。

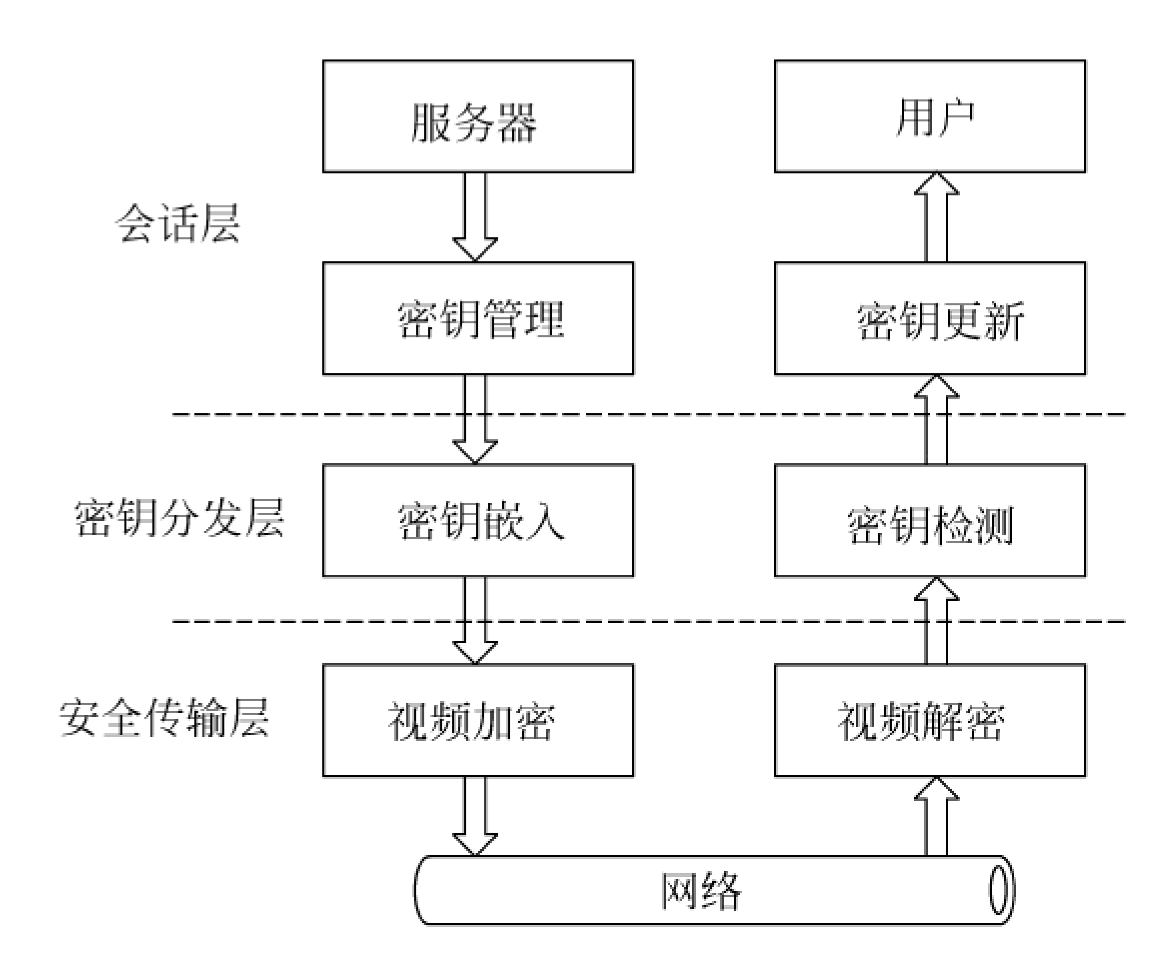


图 5.5.1　MSMP 的层次结构

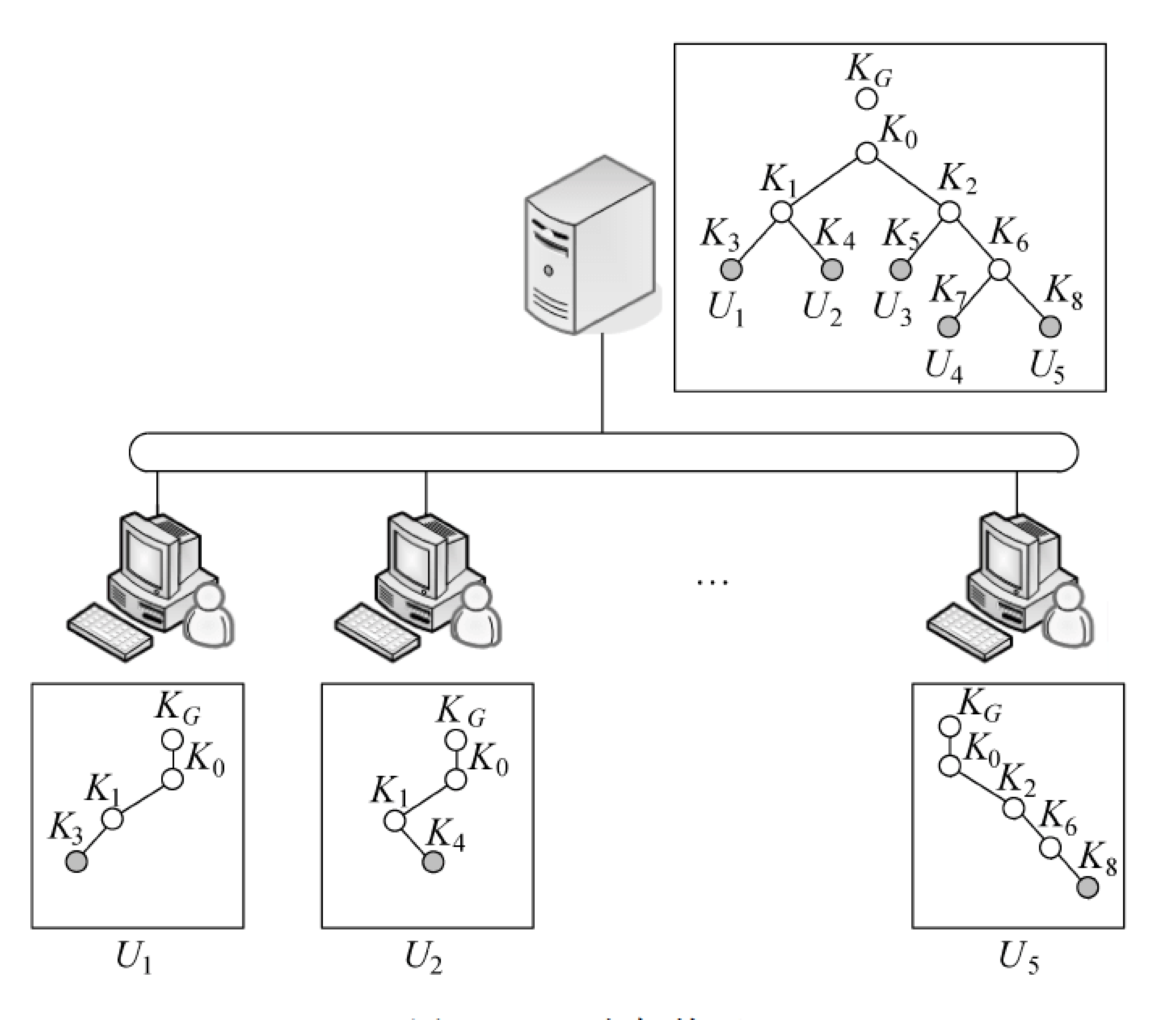


图 5.5.2　密钥管理

在密钥分发层，密钥更新消息被嵌入到视频流中，然后组播给合法用户。在用户端，嵌入的消息被检测出来，然后发送到上层(即会话层)进行处理。为了保证嵌入消息的安全性，携带消息的视频数据的传输实际上是在安全传输层中进行的，如图 5.5.3 所示。在安全传输层，携带了密钥更新消息的视频流通过选择性加密算法进行加密。用户用会话密钥解密然后检测出密钥消息。当所有用户完成密钥更新以后，后来的视频流将会用新的会话密钥进行加密。

考虑到同时会有多个用户加入或者退出，我们将处理时间划分为若干等长的时间片，如图 5.5.4 所示。用户加入或者退出过程完成一系列操作需要的整个时间称为会话周期

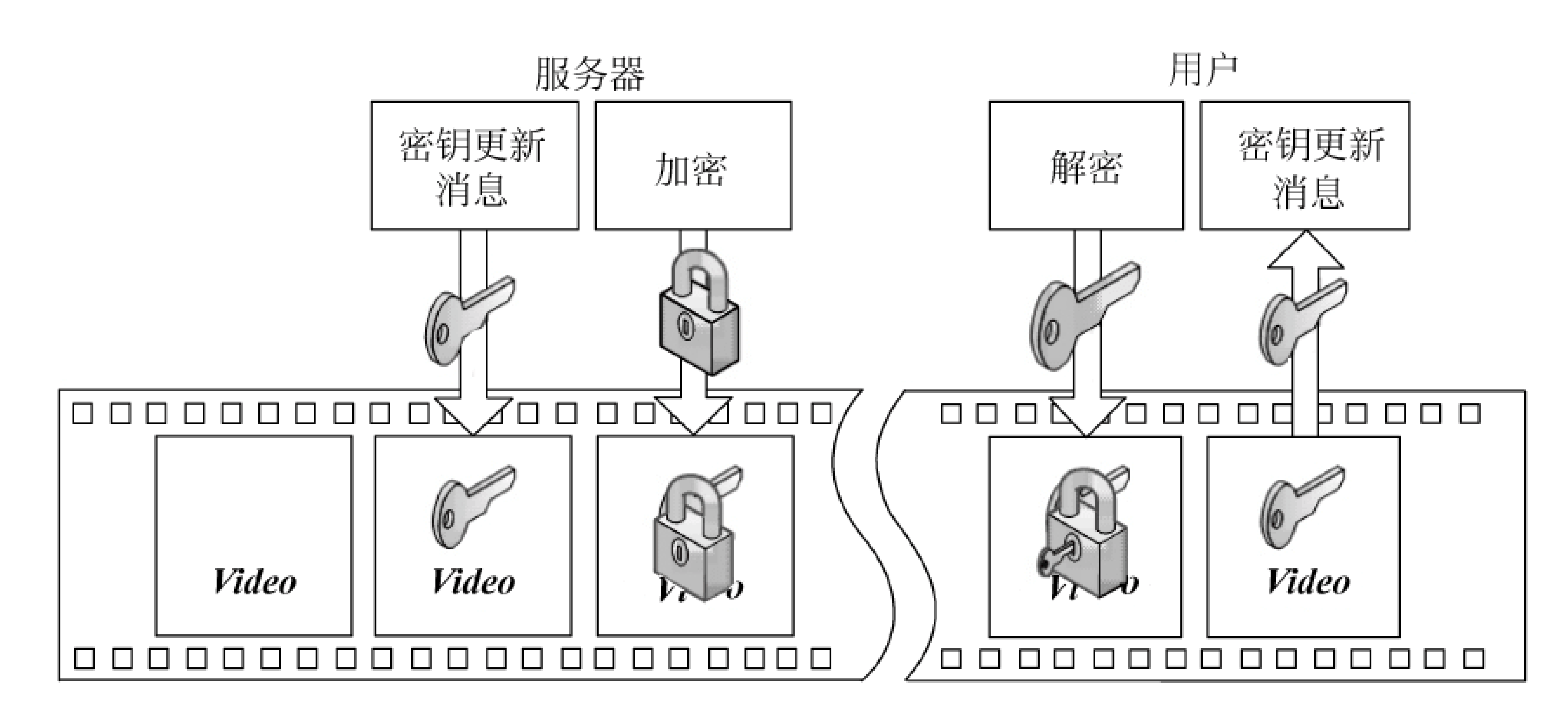


图 5.5.3　安全视频传输

(session period),该周期通常由两个时间片组成。欲加入或者退出的用户在第一个时间片内与服务器联系完成认证过程。同时,服务器修改其密钥树并产生新的密钥更新消息,但是并不立刻启动密钥更新过程。当第二个时间片到来时,密钥更新才真正开始执行,该过程完成以后,用户便完成了登录或者注销过程。此外,在第二个时间片进行密钥更新的同时,服务器能够并行地处理另外一群用户的请求并进行密钥树的维护。这种方法能够避免密钥更新过程频繁地发生,同时又能够提高请求处理的能力。

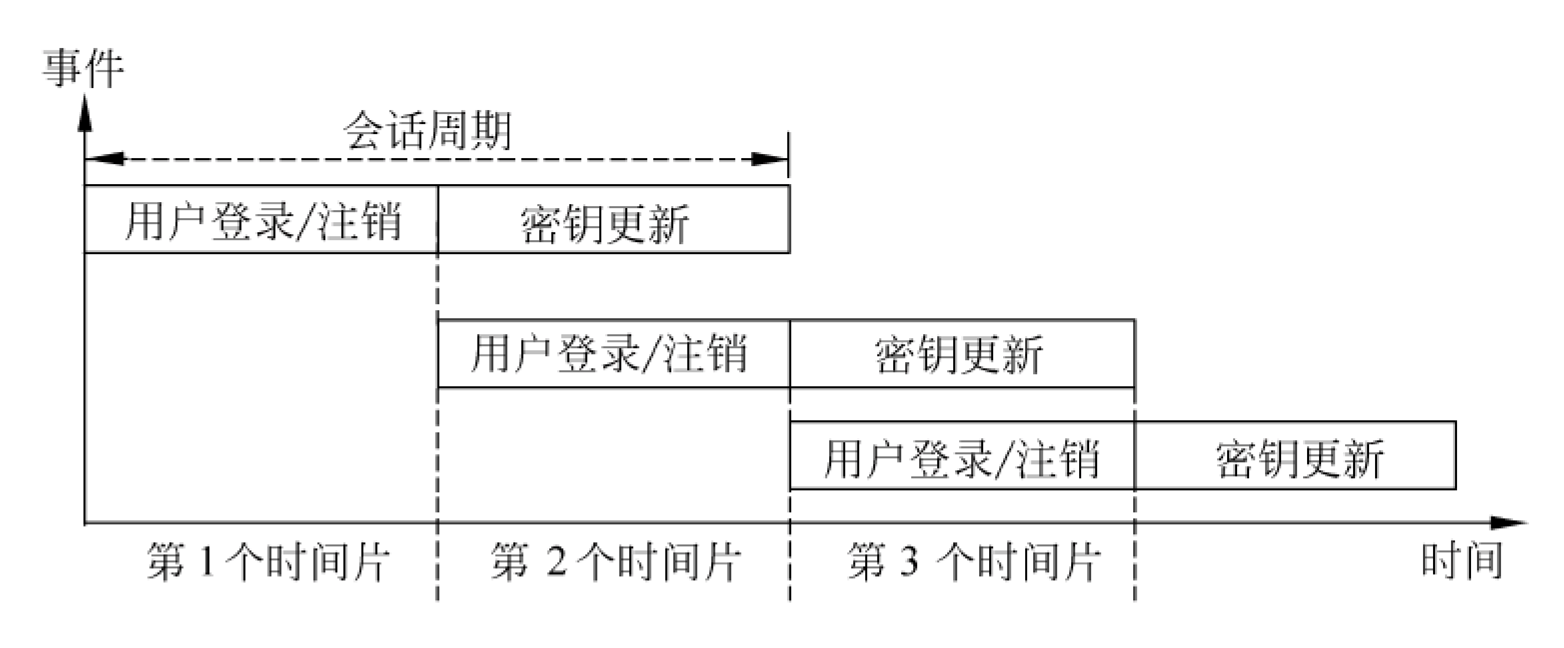


图 5.5.4　时序图

5.5.2　密钥管理与分发机制——LELK 算法

本节基于 ELK 算法提出一种密钥管理和分发方案,称为轻权 ELK(light-weighted ELK,LELK)。尽管 ELK 具有比其他方案更短的密钥更新消息,但其服务器端的计算量仍然很高,特别是在用户退出的时候。由于我们采用的密钥嵌入算法本身提供了有效的差错恢复功能,因此可以去掉 ELK 中的暗示机制以便降低其计算负载。密钥传输的可靠性将由密钥分发层来保证。

5.5.2.1　LELK 的结构

LELK 基于二叉密钥树,扩展了逻辑密钥层次结构(logical key hierarchy,LKH)和单向函数树(one-way function tree,OFT)的方法,是一种高效而安全的密钥分发系统。在 LELK 中,密钥管理服务器(简称为服务器或者 server)负责维护密钥树,该树将用于组播组

的密钥更新。图 5.5.5 是一个简单的密钥树。

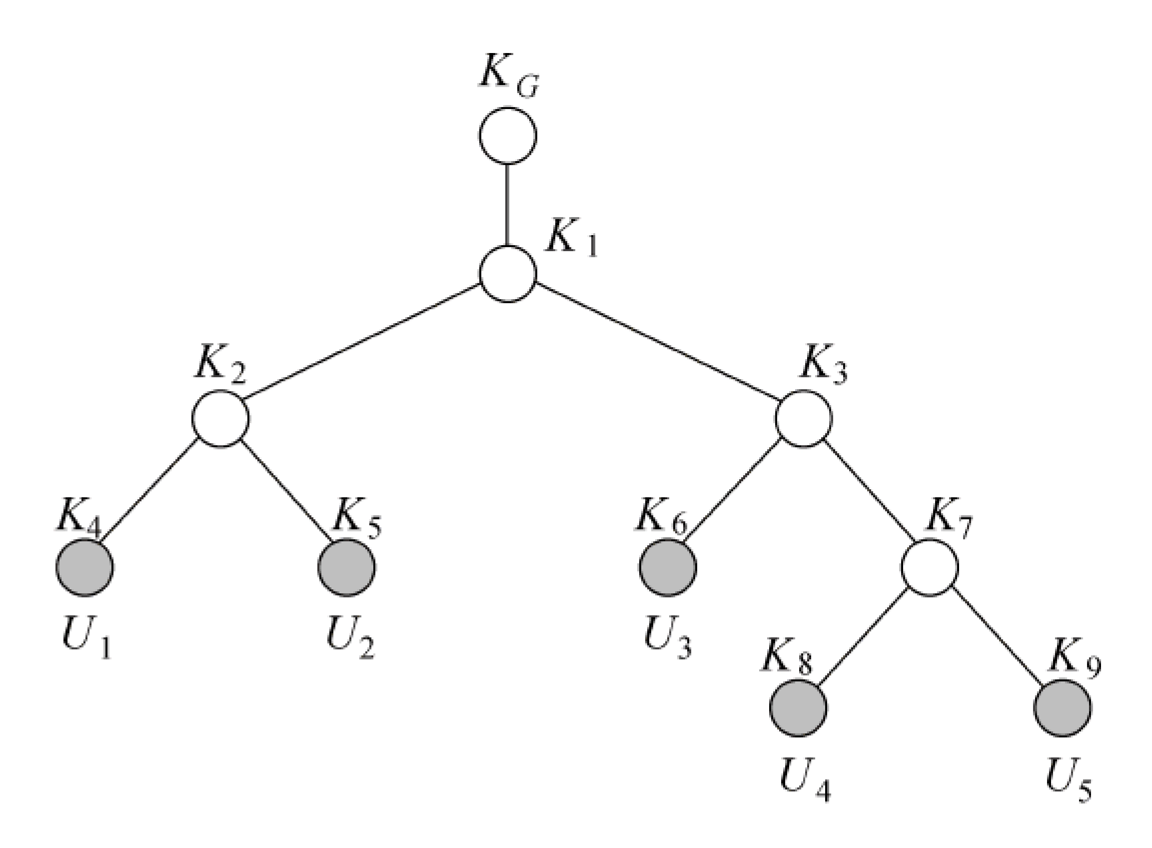


图 5.5.5　一棵简单的密钥树

最上层的节点 K_G 被赋予了会话密钥，该密钥用来加密视频信号。其余的节点都对应于一个 KEK(key encryption key)，该密钥用于加密会话密钥或者其余的 KEK。另外，每个叶节点对应于一个合法用户，或者说，每个合法用户被赋予了一个叶节点。而且，每个用户都知道他所在路径上的所有节点的密钥。如图 5.5.5 所示的例子，用户 U_4 掌握了如下一些密钥：$\{K_G, K_1, K_3, K_7, K_8\}$。

5.5.2.2　符号

文中用到的符号如下：

(1) $\mathrm{PRF}_K^{(m\to n)}(M)$：伪随机函数，见 5.4.4.1 节中的定义。

(2) $\{M\}_K$：表示用密钥 K 对 M 加密。

(3) |：表示将两个序列连接。

(4) $K^\alpha=\mathrm{PRF}_K^{(n\to n)}(1)$，$K^\beta=\mathrm{PRF}_K^{(n\to n)}(2)$，$K^\gamma=\mathrm{PRF}_K^{(n\to n)}(3)$，$K^\delta=\mathrm{PRF}_K^{(n\to n)}(4)$。

注意到(1)～(4)代表了 4 个不同的常量，用作 PRF 的入口参数。这两个函数是为了确保密钥的独立性，也是基于安全性的考虑。

5.5.2.3　LELK 中的密钥更新机制

由于我们采用的密钥嵌入算法能够提供密钥恢复的功能，因此 LELK 中去掉了 ELK 方案中的暗示机制。这使得 LELK 具有更低的计算负载，而同样能够保证密钥更新消息的传输可靠性。

考虑到这样一种情况，若密钥 K 有两个子密钥 K_L 和 K_R，要更新 K 得到 K'，其更新过程需要两个子密钥的贡献。左子密钥 K_L 贡献 n_1 比特，右子密钥 K_R 贡献 n_2 比特，有 $n=n_1+n_2$。密钥更新过程如式(5.5.1)所示：

$$\left.\begin{aligned}&\text{左子节点的贡献值：}C_L=\mathrm{PRF}_{K_L^\alpha}^{(n\to n_1)}(K)\\&\text{右子节点的贡献值：}C_R=\mathrm{PRF}_{K_R^\alpha}^{(n\to n_2)}(K)\\&C_{LR}=C_L\mid C_R=\mathrm{PRF}_{K_L^\alpha}^{(n\to n_1)}(K)\mid\mathrm{PRF}_{K_R^\alpha}^{(n\to n_2)}(K)\\&\text{更新后的密钥为 }K'=\mathrm{PRF}_{C_{LR}}^{(n\to n)}(K)\end{aligned}\right\}\qquad(5.5.1)$$

为了更新密钥 K，服务器必须组播以下加密的密钥更新消息：

$$\{\mathrm{PRF}_{K_{\mathrm{L}}^{\alpha}}^{(n\to n_1)}(K)\}_{K_{\mathrm{R}}^{\beta}},\quad \{\mathrm{PRF}_{K_{\mathrm{R}}^{\alpha}}^{(n\to n_2)}(K)\}_{K_{\mathrm{L}}^{\beta}} \tag{5.5.2}$$

很明显地，密钥更新消息包含了足够的信息。考虑左子树上的用户，他们知道 K_{L}，因此能够自己计算出 C_{L}。但是他们需要 C_{R}，这就是密钥更新消息中含有$\{C_{\mathrm{R}}\}_{K_{\mathrm{L}}}$的原因。同理，右子树上的用户也需要从$\{C_{\mathrm{L}}\}_{K_{\mathrm{R}}}$得到所需的 C_{L}。这样，左、右子树上的用户都得到了足够的计算资料，即 C_{L} 和 C_{R}，然后通过式(5.5.1)中最后一个式子的计算，就能得到新的密钥 K'。

5.5.2.4 用户加入

如果用户加入组播组，密钥服务器会对该用户进行认证并将其分配到密钥树上的一个叶节点上。然后，密钥服务器将会话密钥和该用户所在路径上的所有 KEK 都发送给他。为了保证后向保密性，新用户所得到的所有密钥都应该与以前的密钥无关，应该是服务器随机生成的新的密钥。密钥更新总共有 4 个步骤，如图 5.5.6 所示。

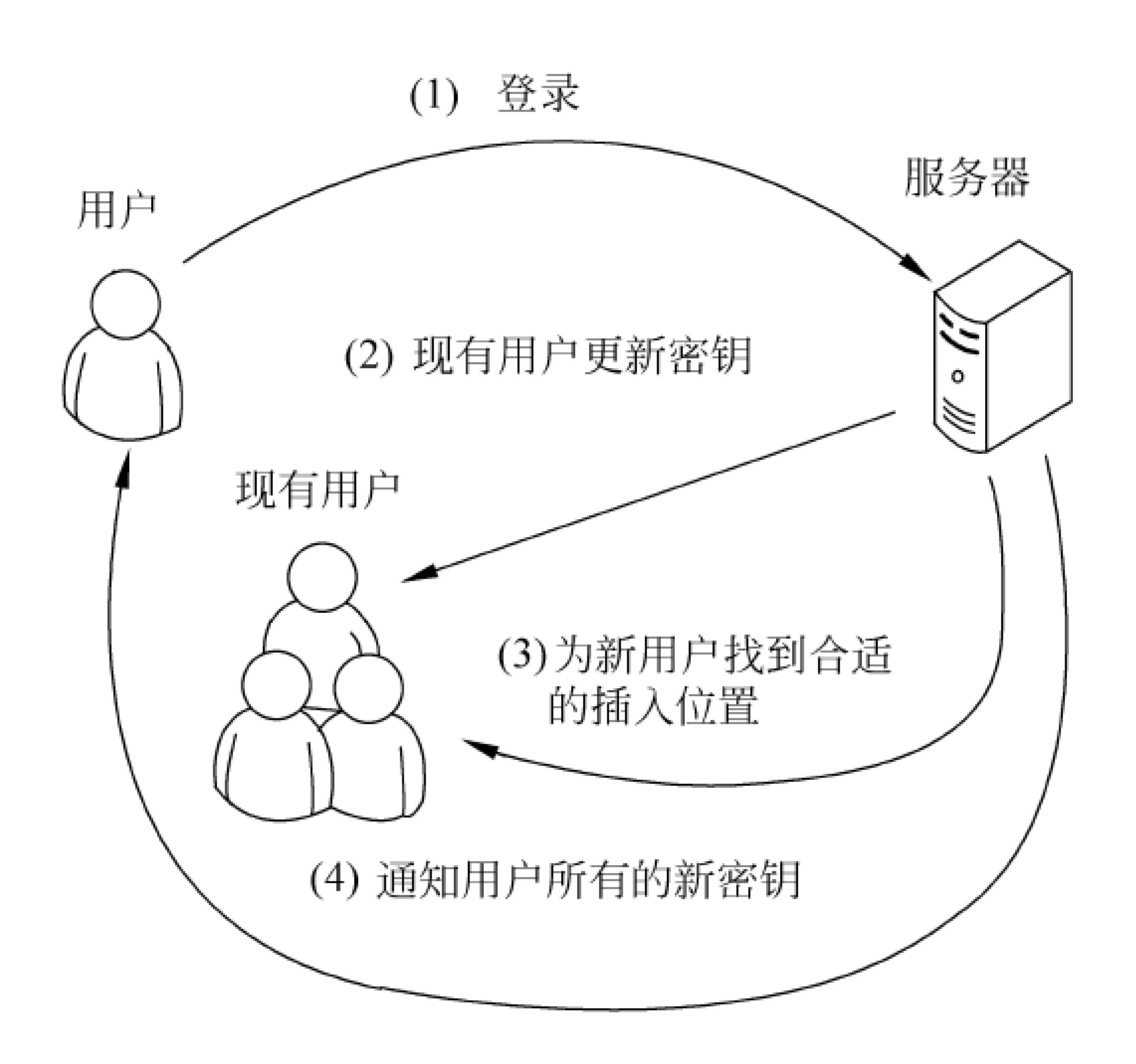


图 5.5.6 用户加入时的处理过程

(1) 新用户(设其标号为 M)与服务器联系。

(2) 服务器组播一条消息，告诉所有在线用户更新其拥有的密钥：

$$\begin{cases} K'_i = \mathrm{PRF}_{K_i^{\delta}}^{(n\to n)}(K_G),\quad K'_G = \mathrm{PRF}_{K_G^{\delta}}^{(n\to n)}(0), \\ K_i^{\delta} = \mathrm{PRF}_{K_i}^{(n\to n)}(\delta),\quad 0\text{ 和 }\delta\text{ 均为参量} \end{cases} \tag{5.5.3}$$

(3) 如果有空的叶节点，服务器便将该用户加入到该节点上，生成一系列新的密钥并发送给他。如果没有空叶节点存在，服务器将生成一个新的叶节点 N_M，并对其赋予新密钥 K_M，该密钥对应于用户 M。然后服务器选择一个叶节点 N_j 来插入节点 N_M。节点 N_j 的选择需要按照如下规则来进行：所选的节点所在路径上的节点数最少。这样选择的目的是为了保证密钥树尽量平衡。假设节点 N_j 上对应的密钥为 K_j，服务器首先将节点 N_j 下移，并生成一个新的父节点 N_P，插到 N_j 原来的位置，而将节点 N_M 插入到 N_P 之下，作为它的另一个子节点。其中，

$$K_P = \mathrm{PRF}_{K_j^{\delta}}^{(n\to n)}(1) \tag{5.5.4}$$

然后，服务器将节点 N_j 加入的消息通过单播的方式告诉相关的在线用户，以便让其了解正确的密钥集。

(4) 服务器将更新后的密钥通过单播的方式发送给新用户。

图 5.5.7 给出了用户 U_6 加入的例子。第 1 步，U_6 首先与服务器联系，服务器更新密钥树中的所有节点。然后，服务器将 U_1 下移，并为其生成一个父节点，对应的密钥为 K_9，并将 U_6 插入到 K_9 之下，同时赋予 U_6 一个新密钥 K_{10}。随后，服务器通知 U_1 增加一个新密钥 K_9。最后，服务器将新密钥 $\{K'_G, K'_0, K'_1, K_9\}$ 发送给 U_6。对于同时有多个用户加入的情况，服务器首先会为这些加入的用户生成一个小的密钥树，然后将这棵树按照上面的方法加入到总的密钥树中。这样，服务器只需要与下移的节点对应的用户进行通信，告诉其位置变化的情况即可。

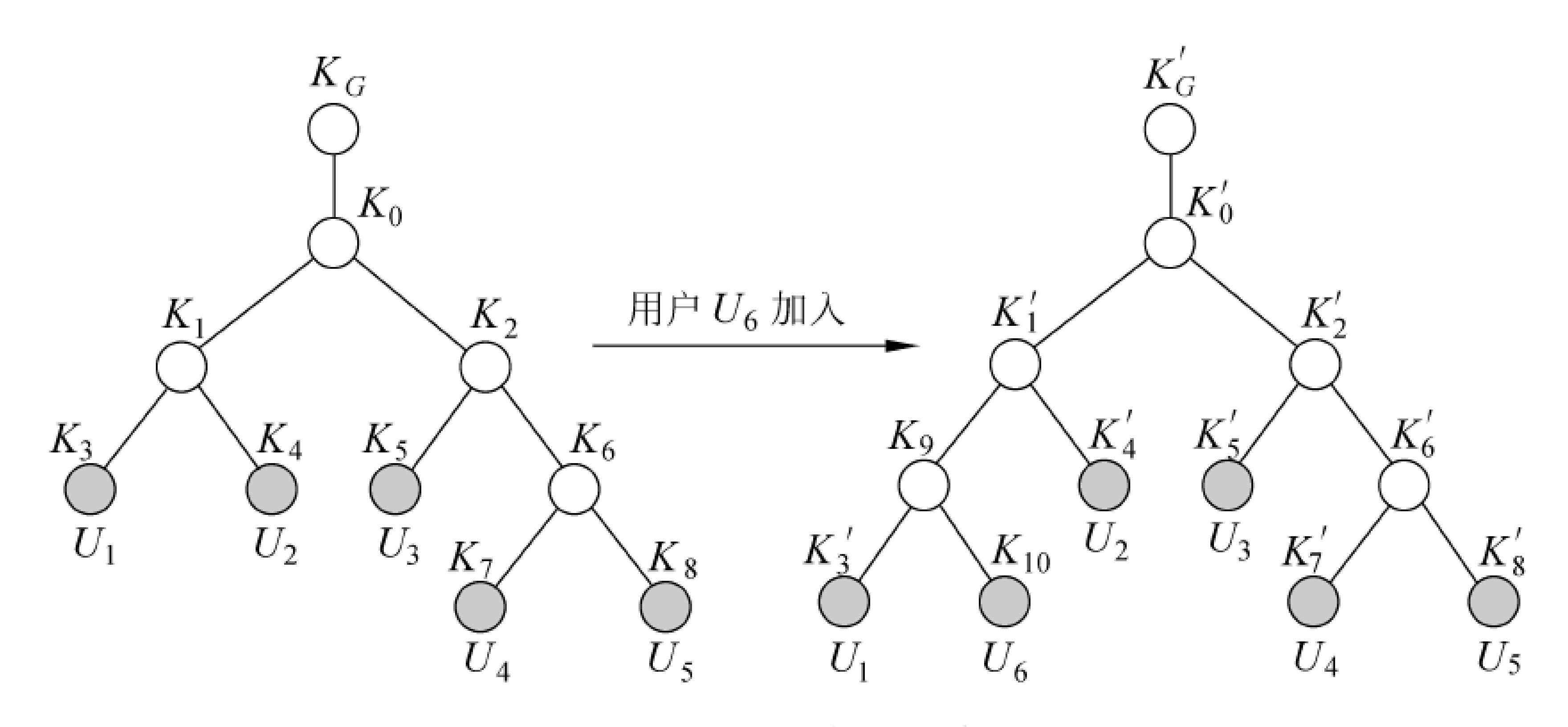


图 5.5.7　用户加入的例子

5.5.2.5　用户退出

用户退出的情况处理起来要困难一些。最主要的问题是如何保证在更新会话密钥时，只有合法用户可以接收到新的密钥，而退出的用户无法接收。实际上，退出的用户原来所掌握的密钥都必须进行更新，以确保系统的前向保密性。密钥更新按叶节点到根节点从下往上的顺序进行。当用户离开时，也需要 4 个步骤的操作，如图 5.5.8 所示。

(1) 欲离开的用户首先与服务器联系。

(2) 服务器删掉该用户对应的叶节点及其父节点，并将其兄弟叶节点上升至原父节点所在位置。然后服务器单播告诉该兄弟节点对应的用户被提升的消息。

(3) 原路径上剩下的节点对应的密钥都要进行更新，更新顺序从下到上。对于每个待更新的密钥 K_i，其计算过程是

$$K'_i = \mathrm{PRF}_{C_{\mathrm{LR}}}^{(n\to n)}(K_i) \tag{5.5.5}$$

(4) 服务器生成所有的密钥更新消息后，开始进行组播，消息如下：

$$\{\mathrm{PRF}_{k_{il}^{\alpha}}^{(n\to n_1)}(K_i)\}_{K_{ir}^{\beta}},\quad \{\mathrm{PRF}_{k_{ir}^{\alpha}}^{(n\to n_2)}(K_i)\}_{K_{il}^{\beta}} \tag{5.5.6}$$

图 5.5.9 给出了用户 U_5 退出的例子。第 1 步，U_5 首先与服务器联系并注销。然后，服务器删掉 U_5 对应的节点，并删掉 K_6，同时将兄弟节点 U_4 提升。接着，服务器计算并更新密钥树中从 U_4 到根节点的所有密钥，并产生一系列的密钥更新消息。最后服务器按顺序组播出 K_2，K_0 和 K_G 的密钥更新消息。例如，要更新 K_2，密钥更新消息是

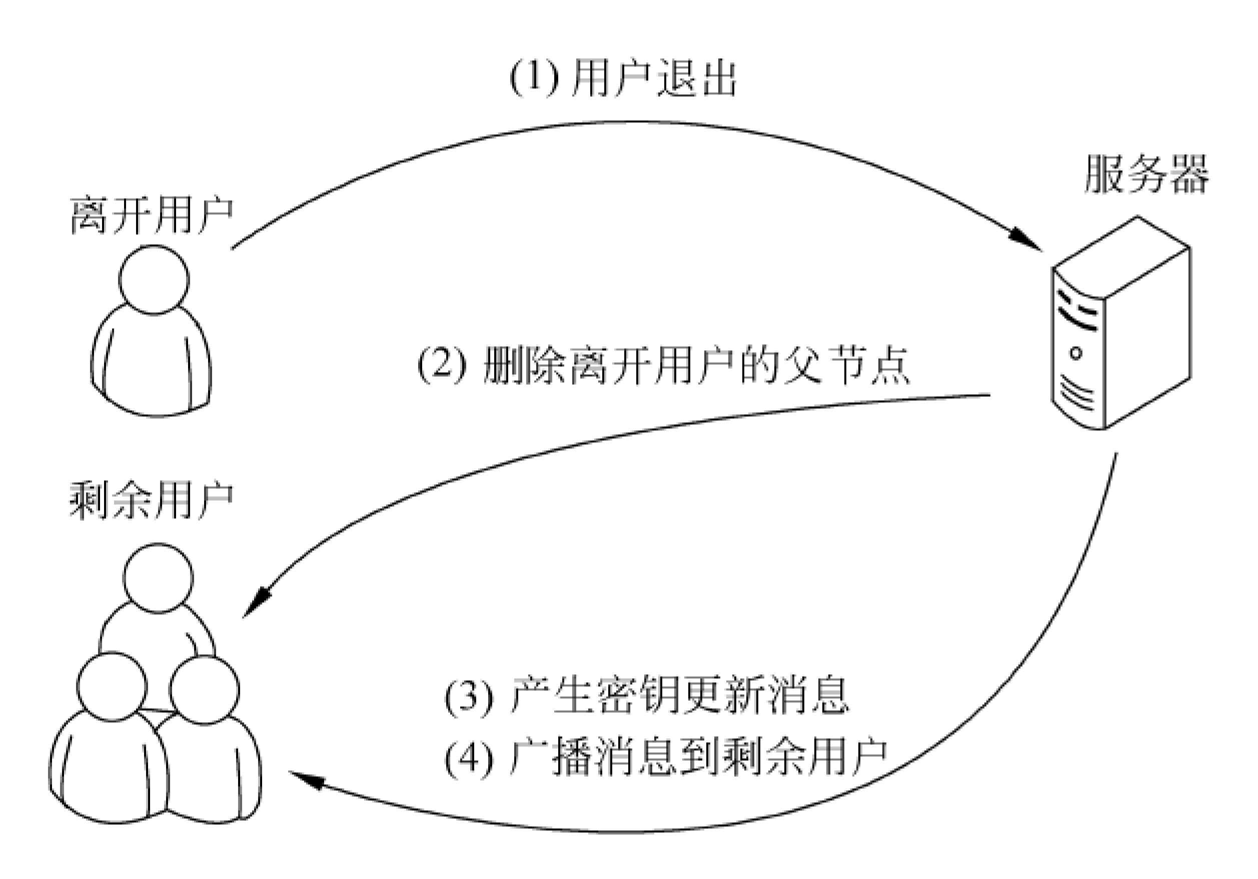


图 5.5.8　用户退出时的处理过程

$$\{\mathrm{PRF}_{K_{\mathrm{L2}}^{\alpha}}^{\langle n\to n_1\rangle}(k_2)\}_{K_{\mathrm{R2}}^{\beta}},\quad \{\mathrm{PRF}_{K_{\mathrm{R2}}^{\alpha}}^{\langle n\to n_2\rangle}(k_2)\}_{K_{\mathrm{L2}}^{\beta}} \tag{5.5.7}$$

用户 U_3 首先计算出 $C_{\mathrm{L2}}=\mathrm{PRF}_{K_5^{\alpha}}^{\langle n\to n_1\rangle}(K_2)$，然后从接收到的$\{C_{\mathrm{R2}}\}_{K_5^{\beta}}$中解密出 C_{R2}。然后计算出 $C_{\mathrm{LR2}}=C_{\mathrm{L2}}|C_{\mathrm{R2}}$，最后得出 $K_2'=\mathrm{PRF}_{C_{\mathrm{LR2}}}^{\langle n\to n\rangle}(K_2)$。

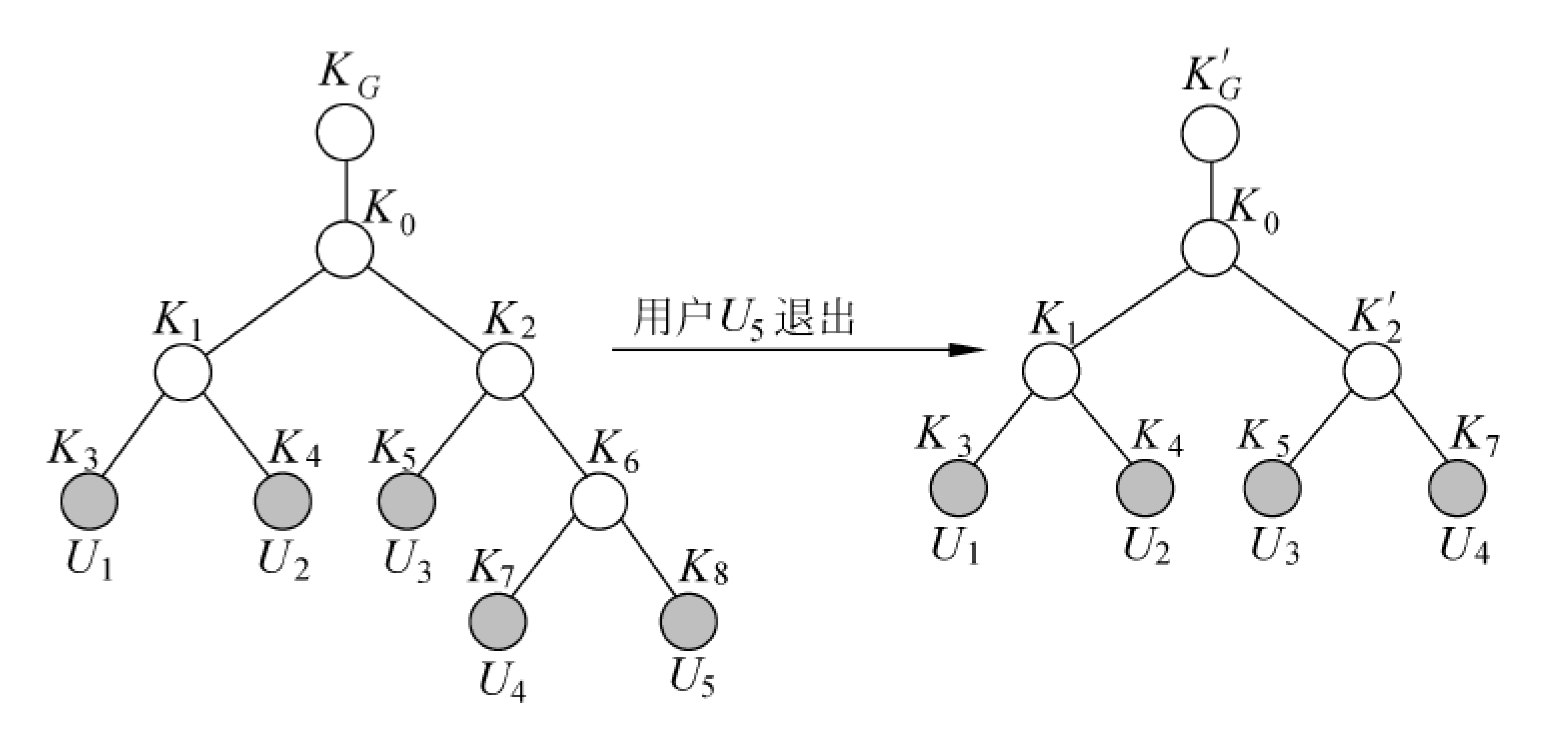


图 5.5.9　用户退出的例子

在同时有多个用户退出的情况下，服务器会逐个修改密钥树，但并不启动密钥消息的分发过程。当这个时间片结束时，服务器会根据最后的密钥树生成相应的密钥更新消息，并组播给所有合法用户。

5.5.2.6　可靠性和扩展性分析

在密钥更新过程中，如果某个组成员未能正确地接收到某个密钥更新消息，那么他将无法解密后续的密钥更新消息，从而无法完成整个更新过程。针对这种情况，密钥嵌入算法已经采用了双重的密钥恢复机制，包括 RS 编码和冗余 GOP，尽最大努力让用户在密钥更新过程结束之前收到正确的消息。如果时间片结束时仍有一些用户没有完成更新过程，那么他们可以直接与服务器联系，通过单播得到更新后的密钥。

当用户离开时，原路径上的所有密钥都需要进行更新。考虑一棵拥有 N 个叶节点的近似平衡的二叉树，每一条路径上的节点数都为 $O(\log N)$。而更新一个密钥需要组播一条消息。因此，组播次数也具有同样的数量级。另外，每条消息长度固定，且都能嵌入到一个图像帧内。根据以上这些特点，可见 LELK 能够有效地面对数量众多的用户带来的挑战。

5.5.2.7　密钥更新消息的格式

前面提到，我们采用的密钥嵌入算法能够在每个 I 帧内嵌入 200 比特的数据。密钥更新消息就在这些数据中，并按一定的格式存放，下面给出这 200 比特数据的格式：

（1）8 位标志，表明数据帧类型等信息。

（2）128 位的密钥更新消息，具体格式见表 5.5.1。

（3）64 位用作 RS 编码的校验码。

表 5.5.1　密钥更新消息的具体格式

位数	用　处	位数	用　处
20 位	密钥 K_i 的序号	32 位	$\{C_{\mathrm{R}}\}_{K^{\beta}_{il}}$
20 位	K_{il}（用于加密 C_{R}）的序号	32 位	$\{C_{\mathrm{L}}\}_{K^{\beta}_{ir}}$
20 位	K_{ir}（用于加密 C_{L}）的序号	4 位	保留

5.5.3　实验分析

在系统的最上层，如图 5.5.10 所示，服务器对密钥进行管理，以确保用户动态变化时的系统安全性。密钥更新消息嵌入到视频数据以后，再经过视频加密处理，最后准备好的视频流通过一些网络协议（如 RTP 和 UDP）组播出去。从前面几节的讨论中我们可以知道，密钥服务器有两个主要的职责，即密钥管理和视频流式处理。在实现过程中，我们将密钥服务器分为两个可信的节点，包括一个视频流服务器和一个网关，如图 5.5.11 所示。

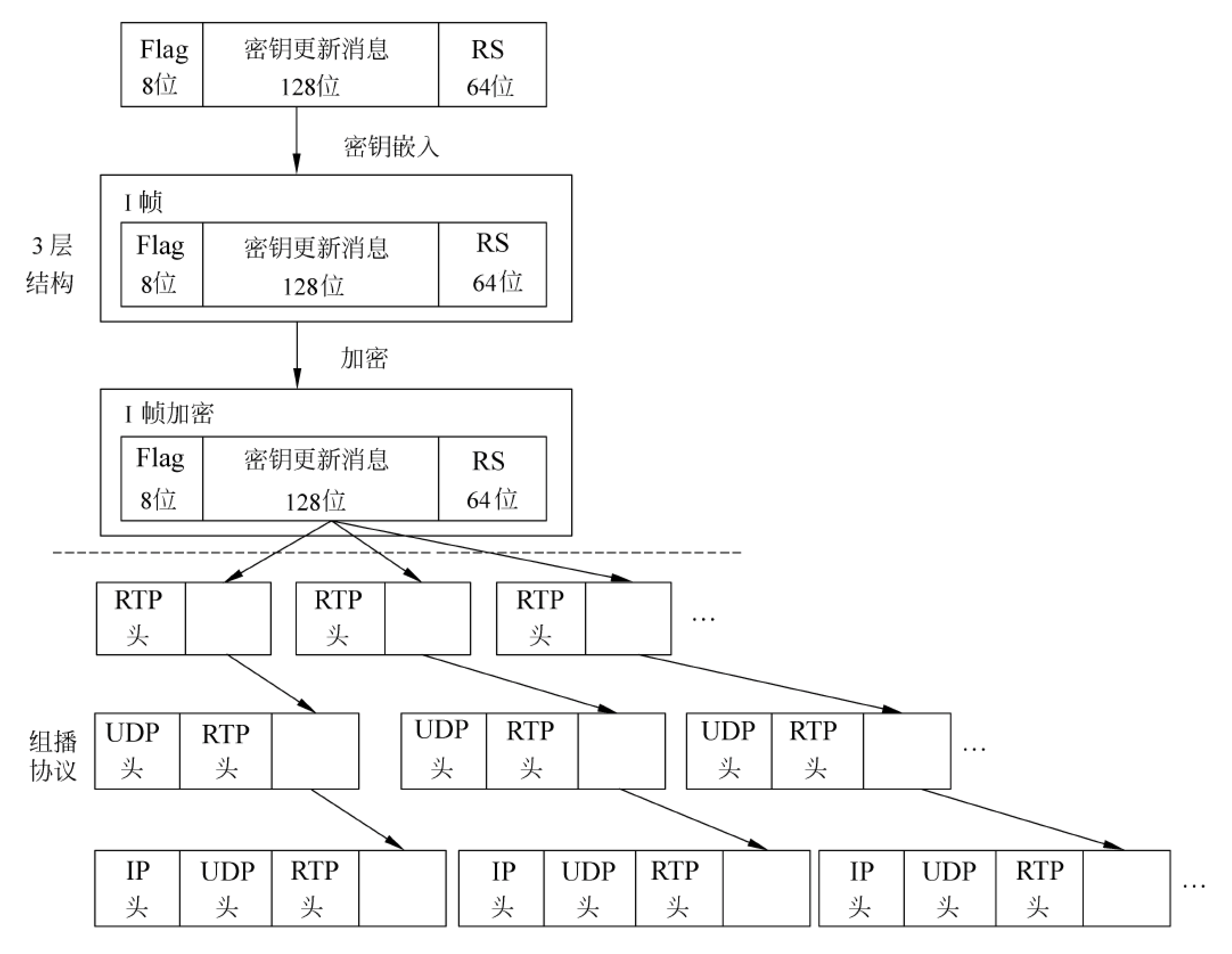


图 5.5.10　系统结构

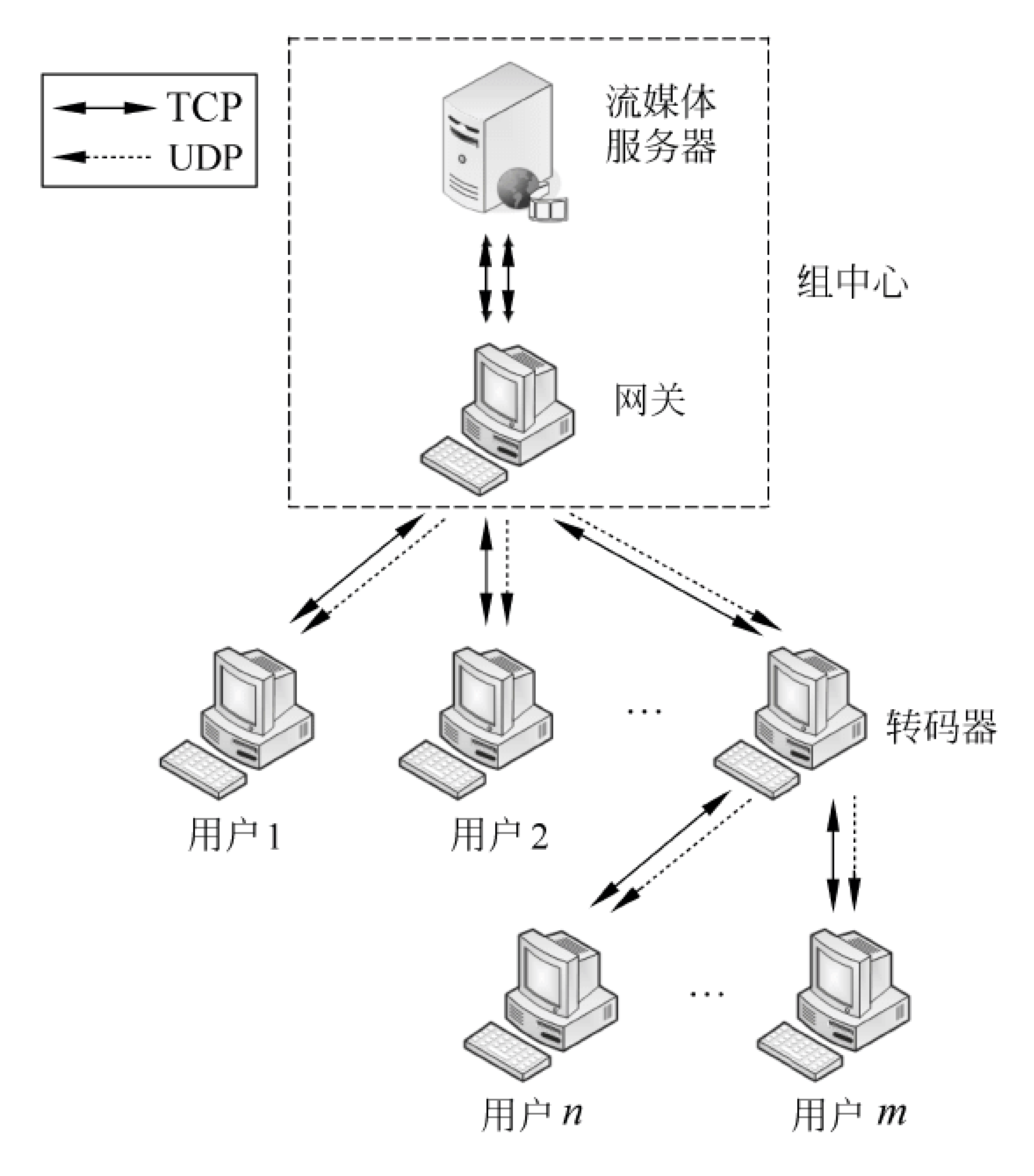


图 5.5.11　密钥服务器的具体实现

当用户事件(加入/退出)发生时,网关与该用户通信,生成相应的密钥更新消息,并通过 TCP 连接将这些消息发送给流媒体服务器。流媒体服务器则负责实时密钥嵌入和实时视频加密。加密的流媒体再次通过另外一个 TCP 连接发送回网关,并由后者以组播的方式转发给所有的用户。采用这两个节点的目的是为了将密钥服务器的这两个主要功能分开,从而平衡负载,为系统的实际运行提供一定的可靠性。

一些自适应的机制(如转码器等)可以视为可信用户。他们同样拥有一些密钥,也接受加密的视频信号。但是在转码以前,视频数据需要被解密。转码结束以后采用同样的会话密钥对视频重新进行加密。然后将新的视频信号组播给相应的用户,如图 5.5.11 所示的用户 m 和 n。这些用户同样要在网关那里注册或者注销,但是其接收到的视频数据不直接来自于网关。

表 5.5.2 给出了在不同规模的组播组中密钥更新实际需要的时间。其中,path_length 是指从某个叶节点到根节点的这样一条路径上最大的节点数。在一棵拥有 N 个叶节点的平衡二叉密钥树中,它满足:

$$\text{path_length} = \lceil \log N \rceil + 2 \tag{5.5.8}$$

假设 T_{GOP} 是一个 GOP(group of pictures,图像组)的长度,常见值为 0.5 s。当用户加入时,服务器会组播一条消息,告诉所有在线用户更新其掌握的所有密钥。这里只需要一次组播,即密钥更新实际需时 $T_{\text{join}} = T_{\text{GOP}} \approx 0.5\ \text{s}$。当用户退出时,必要的消息数量在 $\lfloor \log N \rfloor + 1$ 到 $\lceil \log N \rceil + 1$ 之间(退出用户的叶节点不用更新),因此用户退出时密钥更新实际需要时间为

$$(\lfloor \log N \rfloor + 1) \cdot T_{\text{GOP}} \leqslant T_{\text{leave}} \leqslant (\lceil \log N \rceil + 1) \cdot T_{\text{GOP}} \tag{5.5.9}$$

密钥更新实际需要的时间依赖于组播组的大小和自底向上的密钥更新策略。在实现

中,我们在第1个时间片内等待用户事件(如5.2节提到的),然后在第2个时间片内启动密钥更新过程。这样可以避免在用户事情频繁发生时,不至于反复地进行一些多余的密钥更新过程,其目的在于提高密钥更新的效率和降低网络通信的负载。

表5.5.2 不同规模的组播组实际所需的密钥更新时间

组成员规模	路径长度	加入所需时间/s	退出所需时间/s
100	9	0.5	3.7
1000	12	0.5	5.3
10 000	16	0.5	7.4
100 000	19	0.5	9.0

5.6 流媒体传输的差错控制机制

随着编码技术和流媒体技术的发展,多媒体应用已经扩展到了很多领域,如视频会议、网络电视、视频游戏、远程教育等。越来越多的用户开始接受使用流媒体服务,同时对服务质量的要求也越来越高。如何满足日益增多的多媒体用户的需求已经成为学术界和产业界关注的热点。其中先进的编码技术和有效的数据传输机制是对大规模在线用户提供高质量流媒体服务的关键技术。

本节首先介绍MPEG-4编码标准,然后按照对传输错误处理方式的不同,介绍视频流媒体传输中的4类差错控制机制:信源差错控制、信道差错控制、信源信道联合编码和错误隐藏。

5.6.1 MPEG-4编码标准

MPEG-4是一种基于内容的多媒体数据压缩编码国际标准,是目前使用广泛而成熟的一种视频编码标准。它提供了一个通用的多媒体处理平台,为多媒体通信的发展作出了卓越贡献。目前MPEG-4视频编码标准已经取得了广泛的应用,如Internet视/音频广播、无线通信、静止图像压缩、电子游戏等。

MPEG-采用先进的视频压缩技术,使视频的压缩比率较MPEG-1和MPEG-2标准有了很大的提高。此外,MPEG-4还提供了一种通用的编码标准,能够适应各种网络带宽、各种图像大小、各种图像质量,为用户提供其需要的服务,满足不同处理能力的终端。MPEG-4是一个开放的编码系统,根据不同的需要,可随时加入新的、有效的算法模块。MPEG-4技术已经广泛地应用在如视频电话、视频电子邮件、移动通信、电子新闻等多媒体通信领域。MPEG-4已经突破了传统产业的障碍。同时,MPEG-4在工业界也得到了广泛的支持,如Microsoft,RealNetworks,Apple等主流的多媒体厂商都在各自的媒体格式中兼容了对MPEG-4的支持。

5.6.1.1 MPEG-4的主要特点

MPEG-4 与 MPEG 早期版本有着很大的不同。MPEG-4 不只是具体压缩算法，它是针对数字电视、交互式绘图应用(影音合成内容)、交互式多媒体(WWW、资料撷取与分散)等整合及压缩技术的需求而制定的国际标准。MPEG-4 标准将众多的多媒体应用集成于一个完整的框架内，旨在为多媒体通信及应用环境提供标准的算法及工具，从而建立起一种能够被多媒体传输、存储、检索等应用领域普遍采用的统一的数据格式。MPEG-4 标准支持更丰富的功能，可以概括为以下几个方面：

(1) 基于内容的交互性。MPEG-4 提供了基于内容的多媒体数据访问工具，如索引、超级链接、上下载、删除等。利用这些工具，用户可以方便地从多媒体数据库中有选择地获取自己所需的与对象有关的内容，并提供了内容的操作和位流编辑功能，可应用于交互式家庭购物、淡入淡出的数字化效果等。MPEG-4 提供了高效的自然或合成的多媒体数据编码方法。它可以把自然场景或对象组合起来成为合成的多媒体数据。另外，MPEG-4 提供有效的随机存取方式，在一定的时间间隔内，可以按帧或任意形状的对象，对音、视频序列进行随机存取，或以某个对象为目标在某一序列进行快速检索。

(2) 高压缩率。MPEG-4 基于更高的编码效率，与已有的或即将形成的其他标准相比，在相同的比特率下，它基于更高的视觉听觉质量，这就使得在低带宽的信道上传送视频、音频成为可能。在同等码率的前提下，MPEG-4 标准提供了更好的主观视觉质量的图像。MPEG-4 还提供了对多个并发数据流的编码支持：能够对一景物的有效多视角编码，加上多伴音声道编码及有效的视听同步。在三维视频应用方面，MPEG-4 利用同一景物多视点观察的信息冗余，进而有效地描述三维自然景物。改进后的编码效率和多分辨率数据流编码将使得基于 MPEG-4 标准的应用得到很好的发展。

(3) 通用的存取。MPEG-4 是一个非常开放的系统结构，为了满足不同的应用需求，它提供了大量、丰富的音频视频对象的编码工具，以工具包的形式出现在标准中。在具体实现时，可以根据应用领域的不同，选择使用适当的视频、音频、图形和场景描述工具子集。另一方面，也提高了编解码器的工作效率。框架就是针对特定的应用确定要采用的编码工具，它是所有工具集的一个子集。不同框架的码流句法结构各不相同，而且视频、音频和图形框架中支持的对象类型各不相同。每个框架下又包括有一个或多个级别来限制计算的复杂度。MPEG-4 针对不同的媒体内容和场景描述定义了 4 类框架：视频框架、音频框架、图形框架和场景描述框架。而且，选用不同框架时各部分相互独立。

MPEG-4 的应用范围非常广泛，既可以用于高质量的数字电视，又可以应用于极低码率的移动多媒体通信系统。MPEG-4 编码提供了抗误码机制，能在各种错误易发(error-prone)的环境中使用良好，如无线和移动网络，尤其是在易产生严重错误的低比特应用中。此外，MPEG-4 提供了基于内容的码率分配机制，通过为图像中的各个对象分配优先级，对高优先级对象使用较高的空间或时间分辨率表示，而优先级低的对象分配较低的时间、空间分辨率。基于内容的码率分配是 MPEG-4 的一个重要特性，它提供了自适应可用资源的能力。它允许使用者根据当前系统状态动态地处理不同优先级对象，如对具有最高优先级的对象以可接受的质量显示，次优先级的对象则以较低的质量显示，而不显示其余内容，从而在有限的系统资源内提供更好的服务质量。

5.6.1.2　MPEG-4 系统结构

基于对象的压缩编码是 MPEG-4 的核心，实现基于媒体对象的可交互性、码流的渐进传输都是基于媒体对象处理实现的。在 MPEG-4 标准中，对象是靠层结构表达的。层结构是系统的粘合剂，在系统集成中起着核心作用。分层结构简化了系统结构，能够很好地封装底层技术、规范系统接口及协调系统各部件之间的关系。

1. 媒体对象的树状分层结构

为了实现基于对象的编码，MPEG-4 引入了一个重要概念：媒体对象（media object，MO）。它被定义为有自然语义的编码实体（如行动的人、静止的背景等）。媒体对象是 MPEG-4 的基石，其组织形式的优劣直接影响到编码效率、交互性、可扩展性等性能的好坏。由于各种媒体对象千差万别，所以如何选取一个高效的组织方式成为 MO 表达的关键。

MPEG-4 把各个视频对象（visual object，VO）和音频对象（audio object，AO）按照逻辑隶属关系组合成复合音视频对象（compond audio visual object，CAVO）。多个 CAVO 按照场景描述中的时空关系组合成音、视频场景图。

MPEG-4 采用分层树状结构的目的是为了从 MO 中单独分离出场景描述信息，这些描述信息使得各个媒体对象能够分别独立地解码、重构、组合和重现。用户可以在不解开 MPEG-4 位流的情况下，只用简单替换就可以改变 MO 的组合关系和内容。这对于实现 MPEG-4 基于内容的编辑功能是非常重要的。MPEG-4 通过对媒体对象的分层树状描述既表达了对象间的联系，又使对象间保持相对独立的松耦合关系，为 MPEG-4 实现众多功能打下了基础。当码流传输误码时，媒体对象的逻辑独立性使得可以对发生错误的媒体对象进行丢弃，而不必丢弃整个帧。这使得传输中误码的负作用达到最小。

2. 码流的分层结构

媒体对象的分层树状结构表达需要采用层次码流结构加以支持，MPEG-4 的码流结构如图 5.6.1 所示，在 MPEG-4 的自上而下的层次中，功能由强到弱，结构由复杂到简单。由下至上是层层递进的依赖和封装的关系。

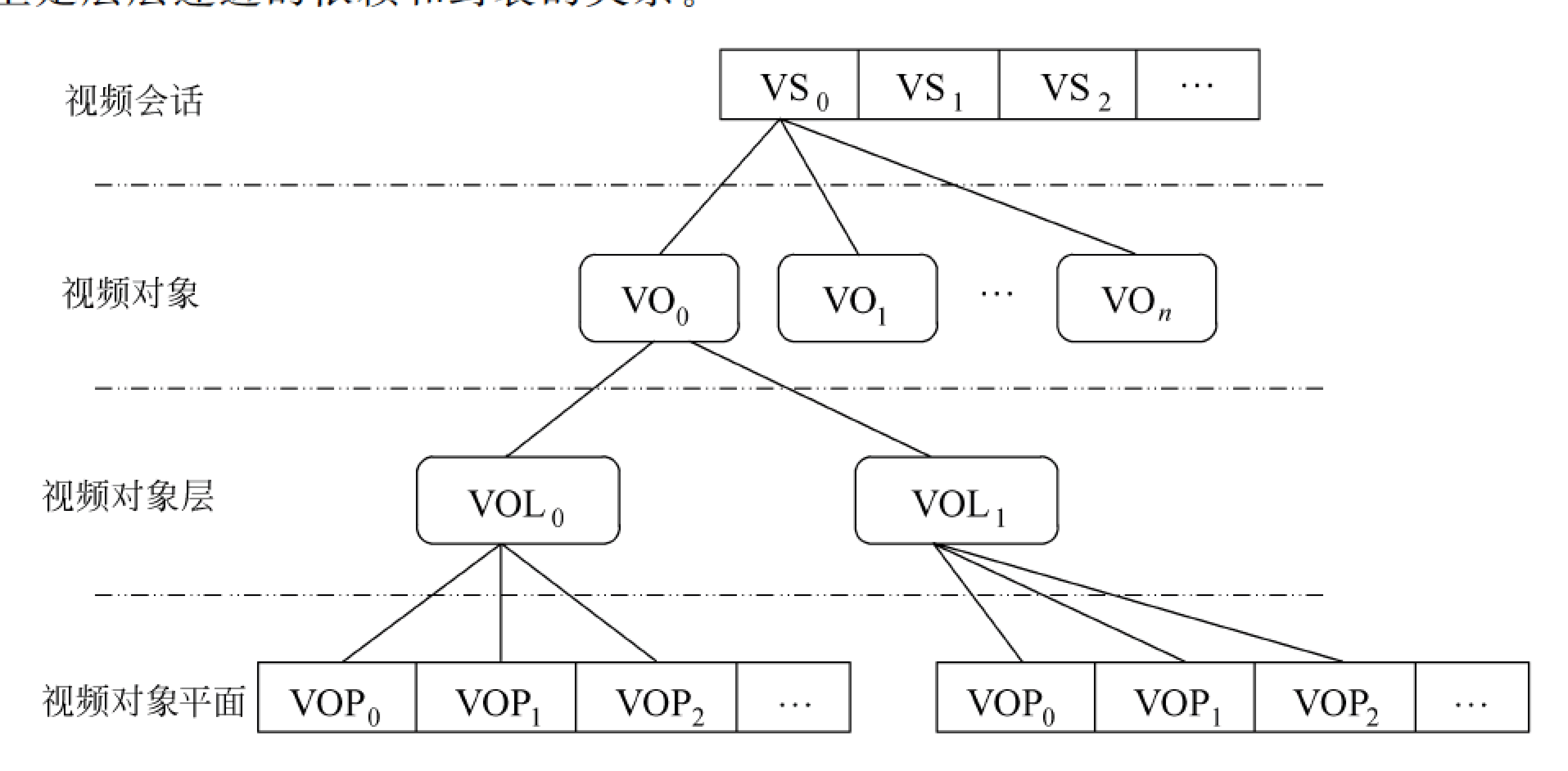


图 5.6.1　媒体对象的树状分层结构

(1) 视频会话(video session,VS)

完整的视频序列通常由多个视频会话组成,它可以分解为多个视频对象。MPEG-4 中规定了组成场景的 4 种标准方式:把音、视频对象放在给定坐标系统中的任意位置;把多个音、视频对象重新组成合成复合音、视频对象;为了修改流式数据音、视频对象的属性,如移动一个对象;交互式地改变用户在场景中的视点/听点。

(2) 视频对象(video object,VO)

场景中的某个物体,是一个三维概念,可以是任意形状。这一层既表现了各种有独立语义的对象,又提供了基于对象的可操作性的最底层。因此,它是整个 MPEG-4 码流的核心层。

(3) 视频对象层(video object layer,VOL)

VOL 分为基本层和增强层:基本层用来传输视频对象最基本的信息;增强层用来根据网络状况或者配置进行视频对象时域或者空域的扩展,提供更好的质量。

(4) 视频对象平面(video object plane,VOP)

它是编码的基本单位,与某个时刻的媒体对象相对应。为与 MPEG-1 和 MPEG-2 相兼容,MPEG-4 也包含了 3 种 VOP:帧内 VOP(I-VOP)、预测 VOP(P-VOP)和双向预测 VOP(B-VOP)。

3. 分层编码

图 5.6.2 为 MPEG-4 分层编码机制。分层编码主要用于 Internet 和无线网等窄带的视频通信、多质量视频服务和多媒体数据库预览等服务。这样就可以对重点对象采用较高的空域或时域分辨率来编码,而对背景之类的不那么重要的对象采用较低的分辨率来编码,从而兼顾了图像质量和带宽问题。MPEG-4 提供了两种基本的分层编码方式:时域分层编码和空域分层编码,且 MPEG-4 支持时域和空域的混合分层。时域分层编码是降低原视频序列的帧率,空域分层编码是降低原视频序列的分辨率。在每类分层编码方式中,视频序列都可以分为两层:基本层(basic layer,BL)和增强层(enhance layer,EL)。基本层提供了视频序列的基本信息,增强层提供了视频序列更高的分辨率和细节;基本层可以单独传输和解码,而增强层则必须与基本层一起传输和解码。空间伸缩性可以通过增强层强化基本层的空间分辨率来实现,因此在对增强层中的 VOP 进行编码之前,必须先对基本层中相应的 VOP 进行编码。同样,对于时域伸缩性,可以通过增强层来增加视频序列中某个视频对象

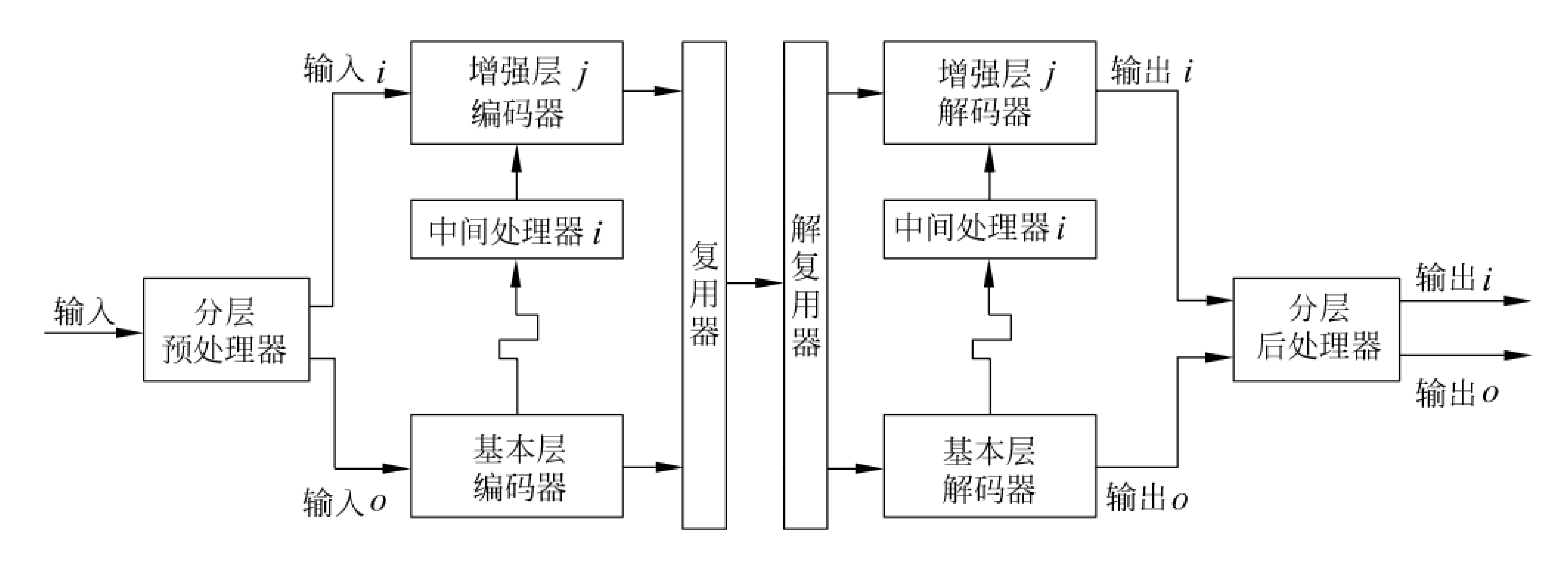


图 5.6.2 MPEG-4 分层编码机制

(video object,VO)的帧率,使其与其余区域相比更为平滑。网络拥挤的时候,MPEG-4 就可以舍弃增强层而只传输基本层,从而可以适应网络资源情况,极大地降低了用户等待的时间。

5.6.1.3　算法概述

MPEG-4 编码标准中针对视频资源,引入视频对象 VO 的数据结构来实现基于内容的表示。VO 的构成依赖于具体的应用和实际系统环境。在要求极低码率的信道条件下,VO 可以是一个矩形帧,这样就与 MPEG-1,H.263 等标准相兼容。基于内容的应用,VO 可能是场景中的某一个物体或某一个层次,也可以是计算机产生的二维或者三维图像。在 MPEG-4 中,VO 作为编码的基本单位,它被定义为画面中分割出来的不同物体。每个 VO 由 3 类信息描述：运动信息、形状信息和纹理信息。

由于视频对象 VO 以视频对象平面 VOP 的方式连续出现,所以针对视频对象编码实际上是对连续的视频对象平面进行编码。MPEG-4 标准的视频编码就是针对这 3 种信息的编码技术。其视频编码器包含对 VOP 的形状编码、运动补偿和纹理编码等基本编码工具。

1. 编解码器结构

MPEG-4 利用复用的方式来实现针对 VO 的视频编码,如图 5.6.3 所示。首先,从原始的视频序列中分割出 VO,再由编码控制机制为不同的 VO 和各个 VO 的 3 类信息分配码率,之后各个 VO 单独编码,最后将各个 VO 的码流复用成一个码流。其中,VO 的分割算法并未在 MPEG-4 中定义。随着相关研究的发展,用户可以选择更好的方法进行分割。在编码控制和复用阶段,可以加入用户的交互控制或由智能化的算法控制。针对 VO 这一层次的视频编解码系统,设计中更多地强调各种 VO 对象或 VO 内部各种数据之间的相对独立性,从而为实现交互式处理提供可能。

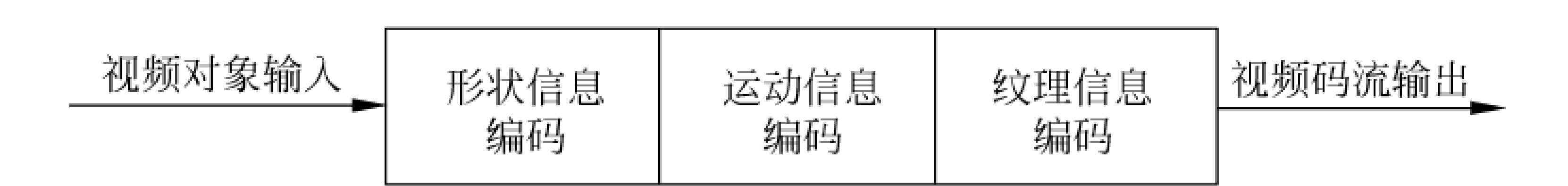


图 5.6.3　MPEG-4 基本视频编码原理

MPEG-4 建立了相应的校验模型(verification module,VM)进行相关的技术测试。如图 5.6.4 所示。VM 将 VO 编码系统进一步细化,按照 VO 的描述分层结构,将 VO 分解成若干个 VOP,然后再分别对每个 VOP 的 3 种信息分别进行编码,最后通过复用形成视频压缩码流。这样,一个具有生命周期的 VO 被划分为在时间上连续的若干个 VOP 帧组成序列,而针对 VO 的 3 种信息的高效压缩编码就主要体现在每个 VOP 编码之中[60]。

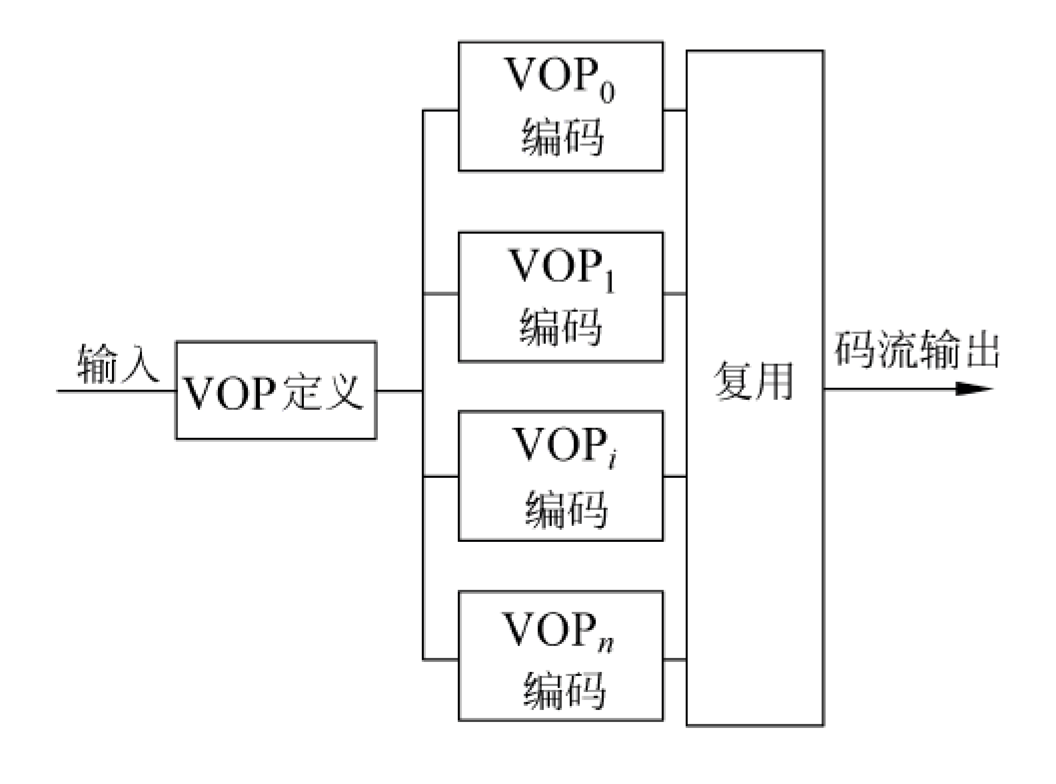


图 5.6.4　MPEG-4 校验模块的基本原理

图 5.6.5 为一个 MPEG-4 编码器的框图。首先获得一个 VOP 的形状信息,然后根据形状信息确定 VOP 区域,通过基于任意形状的运动预测,形成 VOP 的运动信息;再将 VOP 中的所有纹理信息进行编码。最后 3 种信息

经过复用构成 VOP 视频码流数据。相对于其他的视频编码系统,MPEG-4 的 VOP 编码系统最突出的特点是形状编码环节的引入。它是 MPEG-4 基于 VO 编码思想的集中体现。而其运动信息编码和纹理信息编码则更多的是借鉴了成熟的压缩编码技术,并针对 VOP 任意形状的特点进行适当的改进,形成与形状信息结构的新的运动信息和纹理信息编码技术。

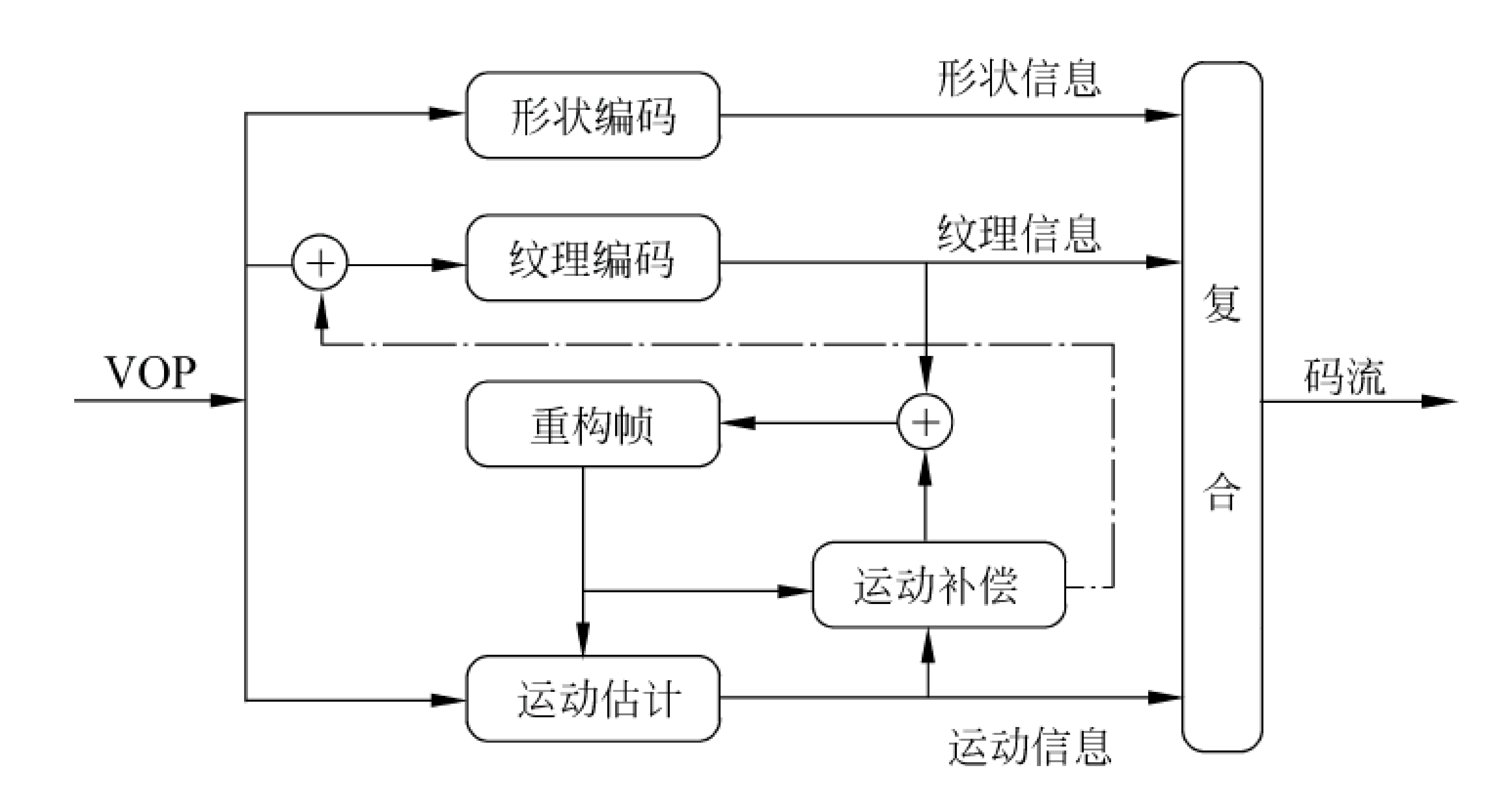


图 5.6.5　MPEG-4 视频编码器的平面结构

2. 形状编码

形状编码作为 MPEG-4 独有的技术,是支持面向视频对象编码的关键所在。形状信息的高效而准确的传递将关系到其他各种基于对象编码技术的实现。一个完整的 VOP 形状信息编码应包括形状信息的生成、形状信息的表示、形状信息的编码等几个相关环节。

MPEG-4 并未规定生成 VOP 形状的具体算法,只是将其列入公开研究的内容。由于 VOP 的定义存在一定的主观性,目前国际上仍缺乏有效的提取算法。现有的 VOP 提取有全自动和半自动两种。全自动方案不需要人的帮助,整个提取过程自动进行。这种方案只适合于已知 VOP 具有特定信息并能与图像的其他区域分开的情况,适用范围较小。半自动方案又可分为两类,一类是重要参数辅助输入,另一类是人工初始输入方案。第 1 类方案依照人对序列和分割结果的判断来调整算法的某些参数,在提取过程之前,需要人工输入运动滤波器和对象跟踪器的有关参数,才能使结果达到最佳。第 2 类方案是通过人工的输入来确定初始帧 VOP 的范围,利用一些算法获得初始帧的 VOP,并在后继帧中利用自动跟踪技术检测这一 VOP 的形变和运动。这类方法的优点是提取 VOP 的边缘比较准确,但不适用于运动视频对象,而适用于静止视频对象,是目前比较成熟的做法。其缺点是用户的工作量比较大,无法实时进行。现有的校验模型 VM 测试序列中,均采用预先确定的 VOP 形状文件来表示 VOP 的形状信息。

视频对象的形状信息有两类:二值形状信息(binary shape information)和灰度形状信息(grayscale shape information)。二值形状信息只用 0 和 1 来表示视频对象平面的形状,0 表示非视频对象平面区域,1 表示视频对象平面区域。二值形状信息编码采用基于运动补偿块技术,可以是无损或有损编码。灰度形状信息用 0～255 之间的数值来表示视频对象平面的透明度,其中 0 表示完全透明。灰度形状信息的引入可以避免前景物体叠加到背景上时,边界过于明显和生硬,形成一种模糊的效果。灰度形状信息的编码采用基于块的运动补偿 DCT 方法,属于有损编码。两种方式相比,二值形式更为简单,灰度级形式更有利于提

高编码对象的主观效果。

在具体形状编码上，MPEG-4 测试模型采用基于位图的形状编码算法，对上述两种形状信息进行编码。这有别于另一种重要的形状编码方法，即基于轮廓的形状编码算法，它主要是记录形状的轮廓信息，更适用于可塑性较高的计算机图形编码系统。

MPEG-4 测试模型 VM18.0[61] 采用基于块的运动补偿和基于块的上下文算术编码(context algorithm encoding，CAE)对二值 VOP 形状信息进行编码，其系统框图如图 5.6.6 所示。VM 在 VOP 形状编码上采用 16×16 的 BAB(binary alpha block) 块作为基本单位，编码算法首先需要对具有任意形状的 VOP 重新确定边界。具体算法利用一个边界框(bounding box)框住 VOP，边框长宽均为 16 的整数倍，同时要求采用最少的 BAB 块的矩形覆盖住 VOP。然后根据编码 VOP 类型、运动矢量和 BAB 块的 ACQ(accepted quality)函数确定 BAB 的类型。从而确定后面的编码方案。MPEG-4 提供了 7 种 BAB 编码模式，编码器需要从这些模式中选取一种作为当前 BAB 的类型。

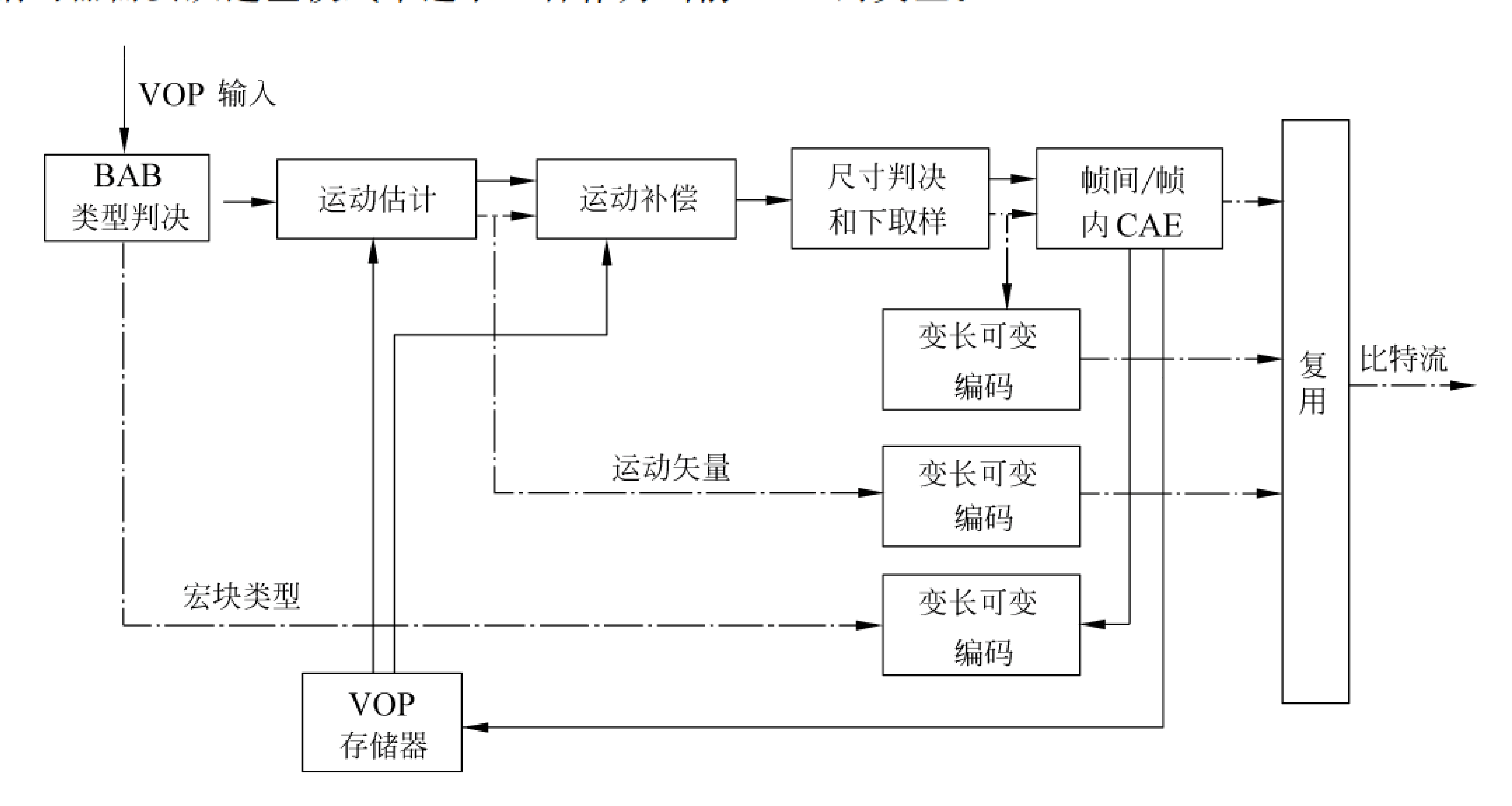


图 5.6.6　二值形状编码系统框架图

对于灰度级形状信息，VM 则综合利用了二值形状信息编码和纹理编码技术，其基本原理如图 5.6.7 所示。在解码端，对 CAE 编码的二值形状解码以后，还需要进行泛化(feathering)，以产生与原来相似的灰度图。

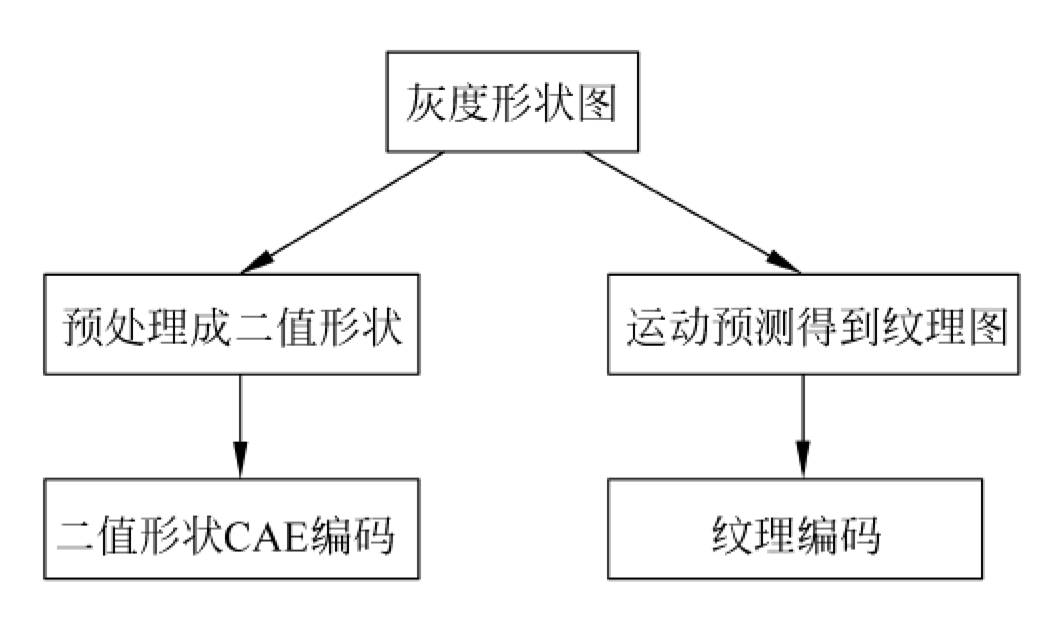


图 5.6.7　VM 中灰度级形状编码原理图

3. 运动编码

运动编码包括运动估计和运动补偿方法，它是 MPEG 系列标准一直采用的一种提高压缩效率和质量的有效方法。其基本思想是考虑相邻帧图像之间的相关性，利用相邻图像对当前将要编码的图像作预测，再将两者之间的差值进行编码传输，以降低码率[62]。

运动补偿实际上是对运动图像进行压缩时所使用的一种帧间编码技术。由于运动的连续性，图像序列中的第 N 帧图像内容的大部分可以看作是前面 $N-1$ 帧图像经过一定的平移操作得到。因此在实际的编码中，为了节省编码比特，并不传输第 N 帧的全部数据，而是利用运动估计技术计算出第 N 帧与预测帧 N^C 的差值 Δ。如果运动估计比较有效，则 Δ 中的概率分布基本上在 0 附近，从而使 Δ 比原始图像 N 的能量集中得多，相应的编码传输 Δ 所需要的比特数也少得多。

在解码端，根据预测帧 N^C 和差值 Δ，就可以基本恢复出初始的第 N 帧图像。这就是运动补偿技术能够去除信源中时间冗余度的本质所在。

运动估计算法是运动补偿过程中的核心算法。它通常被归纳为两大类，一类是像素递归算法；另一类是块匹配算法。像素递归算法基于递归思想，如果连续帧中，像素数据的变化是因为物体的移位引起，算法就会沿着梯度方向在某个像素周围的若干像素作迭代运算，使连续的运算最后收敛于一个固定的运动估计向量，从而预测该像素的位移；而块匹配则是基于当前帧中一定大小的块，在当前帧的前后帧的一定区域内搜索该像素块的最佳匹配块，作为它的预测块。对比较复杂的运动形式来说，尽管像素递归算法的预测精度比块匹配算法要高，但是由于其计算量比块匹配算法大得多，同时块匹配算法本身也具有较好的性能，因此，MPEG 标准中一直使用块匹配算法。

块匹配算法是一种非常直观的运动估计算法，在 MPEG 标准中，它基于物体平移运动的假设来进行运动估计。在平移运动中，物体上的每一点均有相同的速度和方向，在物体运动的轨迹上，当前时刻所处的位置是由前一时刻位置偏移得到的。将块匹配算法运用到上面所述的运动补偿原理当中：当前帧图像被分成二维 8×8 像素的块或者 16×16 像素的宏块，假定每个宏块内的像素都做相等的平移运动，则在其相邻帧中相对应的几何位置周围的一定范围内，通过某种匹配准则，寻找这些像素子块的最佳匹配块。一旦找到，就将最佳匹配块与当前块的相对位移(dx,dy)，即运动向量编码传输。

块匹配算法有多种不同的匹配准则和运动估计方式。匹配准则不仅涉及搜索精度，而且涉及搜索速度。常用的匹配准则有归一化相关函数准则、均方误差准则和平均绝对差准则等。在 3 种准则中，平均绝对差因其计算量小和易于硬件实现的优点而得到广泛应用。

4. 纹理编码

纹理编码的对象是帧内 VOP 中的像素值或帧间 VOP 的预测值。纹理编码主要包括离散余弦变换（discrete cosine transformation，DCT）、量化、DC（direct current）/AC（alternating current）系数预测、反量化、反 DCT 变换、熵编码等部分。图 5.6.8 显示了纹理编码部分的结构框图。

在变换域编码中，KLT 正交变换能够完全去除空间区域内的冗余度，但实现复杂度较高，而且 KLT 变换基函数和区域图像内容有关。DCT 被公认为变换效果最接近 KLT，而且其变换基函数独立于图像内容，所以 DCT 被广泛用于图像压缩，并且是所有基于变换的

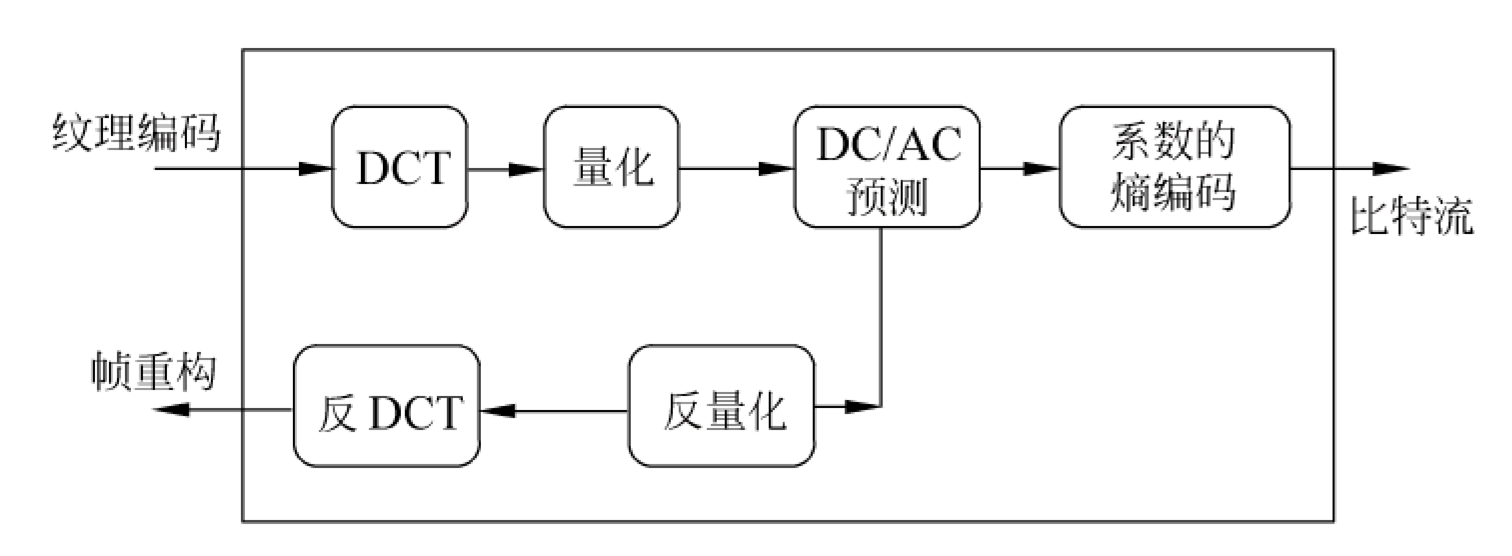


图 5.6.8　纹理编码的结构框架

图像和视频压缩国际标准所选定的基函数。

为了使 DCT 变换的结果能量更加集中，在 DCT 变换之前，先进行低通扩充 LPE（low pass extrapolation）的块填充，从而提高对任意形状区域数据的编码效率。LPE 的填充分为以下 3 步：

（1）计算属于视频对象平面内部区域 R 的像素算术平均值：

$$m = (1/N) \sum_{(i,j)\in R} f(i,j) \tag{5.6.1}$$

其中，N 表示该块位于 R 内的点数。

（2）将 m 赋给块内每个不在 R 中的像素，即

$$f(i,j) = m, \quad ij \notin R \tag{5.6.2}$$

（3）对块内对象区域 R 外部点进行滤波操作，从块的左上角开始沿行一直到块的右下角结束。

$$f(i,j) = [f(i,j-1) + f(i-1,j) + f(i,j+1) + f(i+1,j)]/4 \tag{5.6.3}$$

填充结束后，即可进行 DCT 变换。对于帧内编码方式主要采用 DCT 变换；而对于帧间编码方式，由于在视频对象平面边缘处的宏块有些点不是视频对象平面内部点，不需要进行编码，因此为了减少编码系数，采用形状适应 DCT（shape adaptive-DCT，SA-DCT）变换。

形状自适应 DCT 是 VM 针对 VOP 宏块进行的一种自适应 DCT 变换。对于帧内编码模式的 VOP 边界宏块，由于其内部部分像素不属于该 VOP，为了减少编码系统，VM 中根据形状信息，将宏块内数据进行重组，使得像素值更为集中。在 VOP 重建过程中，由于有形状信息的存在，可以将反变换得到的重建像素数值按照水平和垂直方向移位，最后按照形状信息重新定位，就可以重建该 VOP 内的数据。

MPEG-4 量化环节提供了两种不同的量化过程：H.263 量化模式和 MPEG 量化模式。二者的主要区别在于，MPEG 量化模式引入了非线性量化矩阵，同时在亮度和色度 DC 系数的量化等级选取上也存在一定的差别。MPEG-4 提供可适应性的 DC 预测。DC 系数的预测涉及与当前块相邻的前一个块和上边块的 DC 系数量化值的选择。这种 DC 系数预测的选择方法是基于水平和垂直方向 DC 量化系数的梯度下降原则。如果 AC 参数预测结果比原信号的误差强度更大，则可以禁止 AC 预测。然而对于每一块 AC 预测使能和禁止的切换导致负荷太大，故 AC 预测方式基于宏块切换。

5.6.1.4　容错机制

信道抗误码技术是确保信息有效传递的重要手段。在早期的视频编码标准中，往往只

是直接引用信道纠错编码技术来实现信道保护，例如 H.261 的 BCH 码，但并未从信源编码的角度增强音频、视频压缩数据的抗误码能力。随着多媒体技术向以 Internet 为代表的交互式网络和无线移动网络方向发展，多媒体信息的传递面临着传输信道易受误码干扰的问题。因此，从信道纠错角度无法满足系统稳定性的要求。基于信源编码的抗误码技术正逐步成为多媒体压缩编码中的一个重要环节，如 MPEG-2 中的可分级编码模式，以及 H.263+中针对不同信道特点所引入的多种抗误码编码模式，都是在这一方面的体现。MPEG-4 共提供 4 种容错工具：重同步技术、可逆变长编码、数据分割以及头扩展编码[63]。

1. 重同步

由于现有压缩编码多采用变长编码技术进行统计编码，在消除码流数据元素之间相关冗余的同时，也降低了码流数据抗误码的能力。一个随机比特的误码有可能造成解码器无法正常解码，并且无法正常检测下一个码字的起始位置，导致解码器失步，使得解码过程无法继续。重同步技术的基本思想是在视频码流中的一些特定位置插入特殊的同步码，这种码字在码表中具有唯一性，并且是字节对齐的。一旦出现误码导致失步，解码器则检测下一个同步码，跳过中间的数据，从下一个同步码起重新开始解码。

MPEG-4 标准中的重同步技术类似于 MPEG-2 中的自适应片，也是采用一种视频数据包的重同步结构。算法首先在语法结构上定义一个新的视频包，将每帧图像分割成若干个视频包。视频包由一个个完整的宏块组成，并包括必要的重同步解码头部信息。每个视频包的长度基于它所包含的比特数，如果当前视频包中的比特数超过了预先设定的阈值，就在下一个宏块开始时产生一个新的视频包。MPEG-4 编码器通过每个视频包的起始处插入唯一的重同步码字来实现重同步。如图 5.6.9 所示。此外，为了减少两个视频包中的数据的相关性，在编码的每个视频包的头部除重同步标识外，再插入两个额外的信息：

（1）宏块位置：在视频包中相对于第一个宏块的绝对宏块数；

（2）量化参数：它指明视频包中用来进行 DCT 变换的量化参数。

重同步标识	宏块位置	量化参数	头部信息，运动矢量，纹理数据

图 5.6.9 重同步方法产生的视频包

这种以实际码流编码长度为标准的重同步定位方式，有别于传统的基于空间宏块数量的重同步定位技术。采用这一方式，可以使 MPEG-4 码流具有周期性的重同步标记，避免因为图像内容不同或码率编码的原因导致同步标记不均匀。另外，MPEG-4 还采用一种重同步技术，即固定间隔同步技术。它需要只有在码流允许的、固定间隔位置上的 VOP 起始码字和重同步标记码字。这样可以避免因误码而产生与起始码字竞争的问题。它可以利用固定间隔的特点，判断当前检测的起始码字是码流中正确的数据，还是因误码造成的异常数据。

2. 数据分割

为了有效地定位误码位置，充分利用码流信息，也为其后的误码掩盖算法提供条件，MPEG-4 也采用了数据分割技术。它将每个视频包中的数据重新组织，在头部信息和运动信息与 DCT 系数等纹理信息插入一个 27 比特的分割标识，最后进行传输。如图 5.6.10 所示。

重同步标识	宏块位置	量化参数	运动矢量和头部信息	分隔标识	纹理数据

图 5.6.10 数据分割方法

当视频包中的数据出现错误时，解码器可以分辨出是运动数据错误还是纹理信息有误，从而可以分别进行处理。特别地，当误码被定位在纹理信息区域时，还可以利用已解码的运动信息、结构响应的误码掩盖算法，实现纹理区域的重构。

3. 可逆变长编码

如前所述，在易误码的信道中传输压缩视频通常是变长编码。在解码端，如果解码器检测到数据中的误码，它将失去同步信号直到下一个重同步点，中间的所有数据全部舍弃。而可逆变长编码避免了这个问题，使条解码器通过在误码处的数据反转，以更好地确定误码位置。可逆变长编码使用有前缀特性的特别码字，可以从前向或反向进行解码。这种码字的优点在于，当解码器在前向解码时遇到误码，它可以跳到下一个重同步点进行反向解码，直到遇到误码，基于两个误码的位置解码器可以恢复一些数据。

4. 头扩展编码

解码器在进行比特流解码时需要一些重要的信息，这些信息就是头数据，这些数据包括视频数据的空间维数、解码相关的时间戳和当前图像的编码方式（帧内或帧间）等。由于信道的误码，部分信息可能被破坏，这时解码器只有丢弃属于当前视频帧的所有数据。为了减少对这些数据的敏感性，MPEG-4 引入一种头扩展码（header extension code，HEC），在每一个视频包中，引入了被称为 HEC 比特的 1 比特信息，如果设定了这个信息，那么描述当前视频帧的重要头信息将在视频包中反复出现，通过对视频包中的头信息与视频帧中的头信息进行比较，解码器可以确定视频帧的头信息是否正确，如果是错误的，解码器仍能利用视频包中的头信息对视频帧的其他数据进行解码。在 MPEG-4 的校验模型中，通过使用 HEC 减少丢弃的视频帧的数目，有效地提高了解码视频的质量。

5.6.2 信源差错控制编码

高效的视频编码算法使得视频的流媒体业务成为可能。为了实现高效的压缩编码，目前的视频编码算法使用了运动补偿、帧间预测和变长编码等先进技术，去除了大量的冗余信息，但这导致了视频比特流对传输误码、丢包的高度敏感性。即使是一个误码也可能在一个视频帧中产生大片错误，并扩散到后续帧。另一方面，“尽力而为”的 Internet 无法提供可靠的传输服务，日益广泛使用的无线网、移动网的网络环境更是错误易发，误码率比有线环境严重得多。因此，根据视频编码算法和信道的特点，对传输错误（包括误码和丢包）进行差错控制是视频通信中的一类非常重要的技术，也是近年来视频通信领域的研究热点。

然而，Internet 通常无法提供可靠的多媒体传输服务，造成服务质量的严重下降。虽然 MPEG-4 视频编码标准自身提供了一些容错编码机制，一定程度上控制了传输错误的扩散。一个有效的错误保护机制，将更好地保证多媒体业务的服务质量。

目前的信源差错控制机制主要通过改进码流结构，提高视频压缩码流的抗差错能力，减少错误的发生，或者防止错误的扩散，从而减小错误造成的影响。主要的信源差错控制机制

包括以下几种。MPEG-4 视频编码标准采用了其中的重同步标记、数据分割、可逆变长编码。

(1) 重同步标记

一般情况下,当解码器遇到误码时必须将两个同步标记之间的大量正确数据予以丢弃。如果在两个正常同步标记之间适当地加入特殊的重同步标记,解码器可以借助这些标记重新同步,并大大减少被抛弃的正确数据。

最简单的重同步标记是在视频层次结构的关键位置。这种方法重同步标记之间所表示的图像块的数目是固定的,但内容的不同导致间隔比特流长度不确定。当该区域比特流长度比较长时,重同步标记间隔太远。为了解决这个问题,MPEG-4 中设定了一个阈值,当比特长度超过该阈值时强制插入一个重同步标记。这样可以保证重同步标记的有效作用范围,但它在一定程度上增加了更多的冗余比特。

(2) 数据分割

数据分割技术是将码流中不同属性的数据(如运动矢量和 DCT 系数)分开存放,以便在信道编码时加以不同地保护,有利于保护重要的数据。MPEG-4 也采用了数据分割的方式。

(3) 可逆变长编码

变长编码只允许从正向对码字进行解码,使得错误将扩散到后面连续的码字。可逆变长编码允许解码器对比特流从正向和反向分别解析,能从两个方向逼近错误的位置,从而获得更多的正确数据,缩小了错误的影响范围。

(4) 灵活的宏块排序和冗余片

灵活的宏块排序 (flexible macro-block ordering,FMO)[64]允许在码流的排序中任意排列宏块的次序。其基本思想是将相似的宏块分散排列,如交织条带或分散条带的形式,分成多个片,减少相似数据遭受误码的概率。当遭遇误码时,利用邻近的相关性,采用掩盖技术可以恢复出部分数据。从而提高了视频流的鲁棒性,也降低了编码端为抗误码而采取的帧内刷新的频率,减少了相应的比特消耗。

冗余片 RS(redundant slice)通过对编码片增加少量冗余信息来增加码流的抗误码性。编码时在同一比特流中除基本片数据以外(正常量化值),还包含了冗余数据信息(采用较大的量化值)。当基本片丢失时,可以利用冗余片重建一个较粗的图像。

(5) 多描述编码

多描述编码(multiple description coding,MDC)[65]是将一个视频序列编码成多个相关的码流,并且分别在独立的信道上传送。在接收端,根据被正确传输码流的不同,选择不同的解码恢复方法。只要有一种描述被正确传送到接收端,多描述解码器就可以恢复出一定质量的视频信号。如果有多个描述被正确传输,则视频信号的恢复质量就能得以增强。为了保证从任意描述可以恢复出一定质量的视频,每个描述都必须包含足够多的视频信息。这也意味着多描述编码方案的编码效率要比单描述编码方案低许多。这种损失换来的是对大片误码频繁出现环境下视频信号的鲁棒传输。

一种简单的多描述方案的设计方法是把相邻的采样点分配给不同的信道,形成原始图像的多个子图。为避免突发的连续性误码,可采用交织打包的方法,即相邻子图数据被放入不同的非连续的数据包中,当部分子图的传输受阻时,可用插值的方法利用其他子图恢复原

始图像。但是这种方法必须加入一定的相关性信息，每个子图还要加入头信息。这将进一步降低压缩的效率。

小波变换是传统傅立叶分析发展史上里程碑式的进展，成为众多学科共同关注的热点。它在时域和频域同时具有良好的局部化性质，而且由于对高频成分采用逐渐精细的时域或空域采样步长，从而可以聚焦到对象的任意细节。一些新的视频图像编码标准纷纷采用小波变换代替 DCT 变换，取得了很好的效果。利用三维小波变换实现的可伸缩视频编码方法能够很好地解决网络带宽和随机误码对图像质量的影响问题。

5.6.3　信道差错控制编码

信道传输的差错控制主要由纠错编码器来实现，其基本做法是在发送端对要传输的数据信号按照一定规则附加一些码元，这些码元与原信息码元之间以某种确定的规则约束在一起。在接收端通过检查这些附加码元与原信息码元之间的关系，就可以发现错误和纠正错误。它的任务是构造出以最小多余度代价换取最大的抗干扰性能的“好码”。

信道编码性能指标有几个性能指标：编码效率、编码增益、编码延时、编译码器的复杂度。图 5.6.11[66] 描述了各种信道编码方式的类别关系，这些信道编码广泛用于差错控制技术。信道差错控制技术一般分为自动请求重发(automatic repeat request，ARQ)和前向纠错(forward error correction，FEC)两种基本的误码纠错技术，二者结合起来形成了混合纠错技术(hybrid error correction，HEC)。

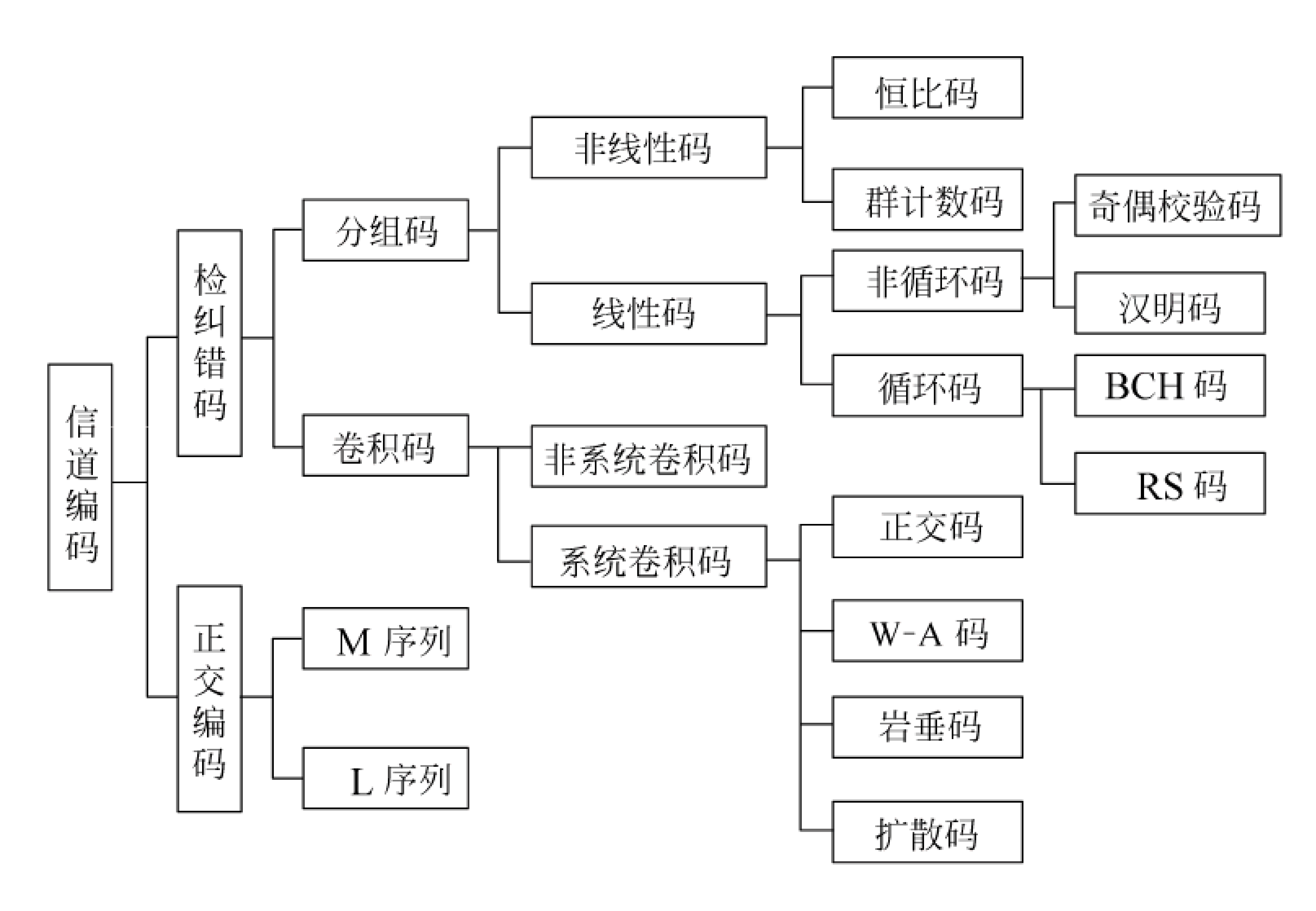


图 5.6.11　信道编码分类树

5.6.3.1　常见信道纠错技术

1. 分组码

所谓分组码是把 k 个信息比特的序列编成 n 个比特的码组，每个码组的 $n-k$ 个校验位仅与本码组的 k 个信息位有关，而与其他码组无关。为了达到一定的纠错能力和编码效率，

分组码的码组长度一般都比较大。编译码时必须把整个信息码组存储起来，由此产生的译码延时随 n 的增加而增加。码组中码元的约束关系为线性，一般用于纠正无记忆信道的突发错误。其中二值的 BCH 码和非二值的 RS 码是应用最广泛的线性循环分组码。

BCH（Bose Chaudhuri Hocquenghem）码是循环码的一个重要子类，它具有纠多个错误的能力，是目前研究比较成熟的一种纠错码。循环码建立在严密的代数学理论基础上，而且编码和解码设备都不太复杂，检纠错能力较强，所以这种码得到了广泛的应用。循环码最显著的特性是循环性，即循环码中任一码组循环一位仍为码中的一个码组。它的生成多项式与最小码距之间有密切的关系，人们可以根据所要求的纠错能力很容易地构造出 BCH 码，它们的译码器也容易实现。RS（Reed-Solomon）码是一种非二值的 BCH 码，它在伽罗华域 GF（Galois field）中进行运算。RS 码的编码过程就是计算信息码组多项式 $M(x)$ 除以校验码生成多项式 $G(x)$ 之后的余数。

在伽罗华域 GF($2m$)中，RS(n,k,t)的符号含义如下：

m：表示码组由 m 比特组成；

n：表示码组总长度；

k：表示码组中的信息长度；

t：表示能够纠正 t 个码组的错误，即 t 个 m 位的二进制错误码组。

其中，$n-k=2t$ 表示监督码组个数。对于一个 m 位的二进制错误码组来说，只有一位错误还是 m 位全错不影响错误恢复，因此 RS 码特别适用于存在突发错误的信道。

2. 截短码

大部分循环码都存在自身的一种缩短形式($n-s,k-s$)，称为截断码。实际应用中，循环码可能需要不同的码长，我们可以从($2m-1,k$）码中挑出前 s 位为 0 的码组构成新的截短码，这种码的监督码位数不变，因此纠错能力保持不变，编码效率提高，但是没有了循环性。截短码给信道编码带来很大的灵活性。

3. 卷积码

卷积码属于非分组编码，编码器具有记忆性，主要用于纠正随机错误。卷积码是将 k 个信息比特编成 n 个比特，但 k 和 n 通常很小，适合以串行形式进行传输，产生的时延很小。与分组码不同，卷积码编码后的 n 个码元不仅与当前段的 k 个信息有关，还与前面的 $N-1$ 段信息有关。因此编码过程中互相关联的码元个数为 nN。卷积码的纠错性能随 N 的增加而增大，在编码器复杂性相同的情况下，卷积码的性能优于分组码，但卷积码没有分组码那样严密的数学分析手段。

4. 打孔码

打孔码（rate-compatible punctured convolutional codes，RCPC）[67] 是一种演变的卷积编码，特点是编码器由一个母卷积码编码器和一个删除器组成，同一组 RCPC 编码器的母卷积码编码器完全相同，仅删除器所使用的删除表不同，通过删除表的不同可以灵活地调节编码码率，实现不同程度的差错保护，因此 RCPC 码经常与 ARQ（automatic repeat request，自动请求重发）配合使用，根据 ARQ 的反馈信息调整删除表，实现差错保护粒度的自适应变化。

5. Turbo 码

当交织长度足够长时，Turbo 码具有接近 Shannon 极限的优越性能。大量的计算机仿真和不同码结构研究表明，虽然 Turbo 码译码复杂度要大于传统的卷积译码，并且有较大的时延，但在无线信道低信噪比的情况下，Turbo 码的性能要优异得多。Turbo 编码的基本思想是两个交错并行的卷积编码，如图 5.6.12 所示，编码器 1、2 可为卷积编码或者分组码，同时也可推广至多维。

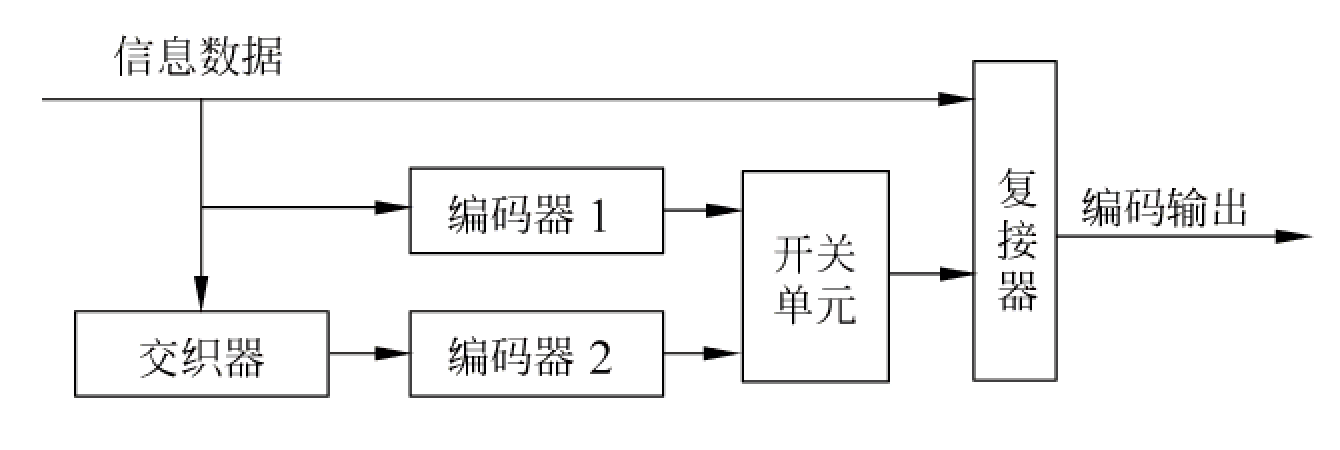


图 5.6.12　Turbo 编码框架结构

6. 交织技术

在错误易发信道中，比特差错经常是成串发生的，然而信道编码通常仅在检测和校正单个差错和较短的差错串时才有效。因此，将一条消息中的相连比特分散开，即一条消息中的相连比特以非相连方式被发送，能够很好地缓解这个问题。这种方法就是交织技术。从本质上说，交织器的目的是使信道传输过程中所突发产生集中的错误最大限度地分散化。

交织器包括规则交织器、不规则交织器、随机交织器三大类。规则交织器又称为分组交织器，也就是行读列出或列读行出的交织器；不规则交织器目前主要有对角交织器、螺旋交织器和奇偶交织器等形式。对角交织器和螺旋交织器都是采用行写而对角读出的方式。

奇偶交织器不是一种独立的交织器生成方法，而是配合删除技术在交织器生成时加上限制条件的一种方法，这里的删除技术是在编译码过程中将信息以删除截短码的形式送入信道，收端通过加入模拟零的方式加以恢复，这个过程降低了码元的纠错能力，但却能提高编码效率，Turbo 码就是运用了这种交织加删除器的技术，它会在生成截短 Turbo 码的同时每次分别将经由编码器对应于输入的奇数和偶数信息位所产生的冗余位交替地送往信道，这样来保证信息序列中的每一位均有对应的冗余位通过信道传输送抵译码器。

随机交织器对于每一组信息序列所产生的交织结果是随机的，所以在传输编码序列的同时，在信道上还要传输交织器的信息，这不仅加大了译码器的复杂度，而且也加大了信道负载，所以现在采用的随机交织器都是伪随机的，即事先经过随机选择而生成一种性能较好的交织方式，然后将其做成表的形式存储起来进行读取。

5.6.3.2 自动请求重发

在自动请求重发(automatic repeat request，ARQ)机制中，发送端的数据中携带一定数量的纠错码，当接收端检测出错误时就通过反馈信道通知发送端重发该码字，直到正确接收为止。

ARQ 的优点在于编译码设备简单，在一定冗余下检错码的检错能力比纠错码的纠错能力要高得多，因而整个系统的纠错能力极强，可以获得极低的误码率。此外，由于检错码

的检错能力与信道干扰的变化基本无关,使得这种系统的适应性很强,特别适用于高误码率的环境。另一方面,ARQ 需要反馈信道,适用于点对点的通信,要求收发两端必须相互配合、密切协作,因此这种方式的控制电路比较复杂。若信道干扰很频繁,则系统经常处于重发消息的状态,因此这种方式传送消息的连贯性和实时性较差。在视频通信中,通常允许一定程度下的图像延迟,因此可以利用反馈信道来传输出错信息,从而改变编码策略来提高抗误码能力。在很多基于 H.263 的实时视频传输系统中,往往通过反馈信息来控制周期性的帧刷新,每隔一段时间采用 I 帧编码模式来防止误码的扩散。图 5.6.13 描述了 ARQ 机制的系统框架。

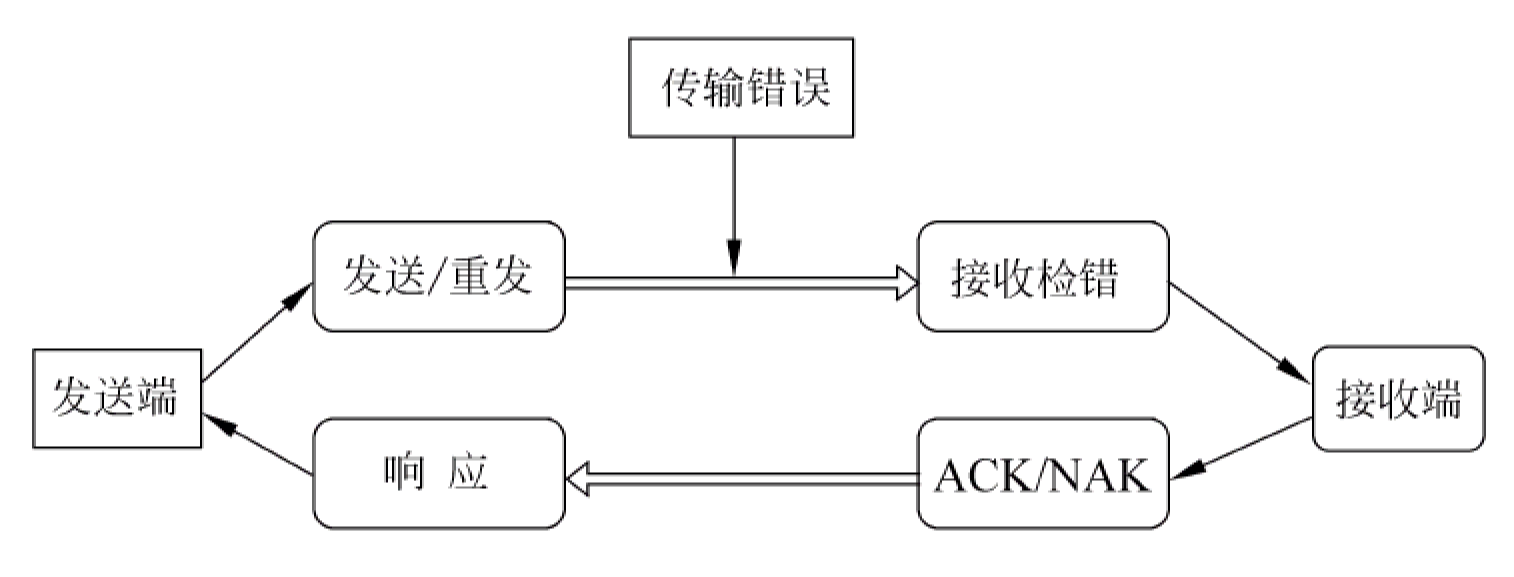


图 5.6.13 ARQ 机制的系统框架

基本 ARQ 方案包括停待式 ARQ(stop-and-wait ARQ,SAW ARQ)、退 N 步 ARQ(go-back-N ARQ,GBN ARQ)和选择重传 ARQ(selective-repeat ARQ,SR ARQ)三种。一个 ARQ 通信系统性能的评价指标包括吞吐效率(单位时间内接收系统接收并传送给用户的数据组的平均数与发送端发送的平均组数的比值)和可靠性(通信系统在规定条件下完成信息正确传输的能力)。

5.6.3.3 前向纠错

前向纠错(forward error correction,FEC)方式不需要反馈信道,能够进行一对多的组播通信,实时性比较好,因此被广泛应用于基于 IP 的组播通信,能够保证所有的接收用户即使收到内容的不同部分,也可以正确地进行译码,可以减少甚至完全消除接收端向发送端发送关于类似于丢包的反馈信号,避免了建立反向信道的麻烦。

经典信道 FEC 编码[68]的基本思想是将 k 个源数据包通过 FEC 编码生成 h 个冗余数据包,冗余数据包与源数据包一起组成包含 $n=k+h$ 个数据包的传输组。数据包以传输组为单位,通过网络传输给接收者。若发生丢包和误码,传输组中的部分数据在传输中会丢失或发生错误,只要发生错误的数据包小于或等于 $n-k$,即接收端只要接收到其中任意 k 个数据包,就可以进行解码恢复出所有源数据。

FEC 通信系统的优点是只有一个信道,而且系统的传输效率高,特别适合于流媒体一类对实时性要求较高的数据传输。然而 FEC 也有一些缺点:如果信道错误比特率高于 FEC 所能提供的纠错能力,那么 FEC 编码不能起到任何作用;反之,如果错误比特率较低,FEC 编码的作用不明显,当译码错误时,错误的信息也送给用户,降低了通信系统的可靠性。要获得较高的系统可靠性必须使用长码和选用纠错能力强的码组。这使得译码电路复杂化,造价提高。

5.6.3.4 混合纠错方式

鉴于 FEC 与 ARQ 系统各自的优缺点，适当地把它们结合起来，就构成混合纠错(hybrid error correction，HEC)[69]通信系统。HEC 系统的可靠性比 FEC 系统要高，传输效率也比 ARQ 系统高，因此在分组数据交换网或计算机通信网中人们更愿意使用这种混合式差错控制系统。

目前常用的 HEC 类型有 3 种，分别为Ⅰ型 HEC、Ⅱ型 HEC 和Ⅲ型混合 HEC。

Ⅰ型 HEC 系统在发送端有效信息的基础上增加具有检错和纠错能力的冗余码字，如果接收端接收到的错误码字可以被纠正，则送给接收用户；否则，接收端发出重传请求，请求重传同样的码字，直到重传码字能够被正确接收或者超过重传的次数为止。

在Ⅱ型混合 HEC 系统中，除信息比特 I 外采用两个线性码：一个高速率的(n,k)码 C_0，仅用于检错，一个半速率可逆$(2k,k)$ 码 C_1，用于同时纠错和检错。数据包第一次发送时只进行检错编码(即数据包内含 I 和 C_0)，当接收端发现接收到的数据包有错误时，一方面将接收到的误码存储于寄存器中，另一方面发送重传请求给发送端。发送端接收到某一数据包的重传请求后将发送由 I 产生的纠错码 C_1，接收端接收到 C_1 后，用它对寄存器中的误码进行纠错。如果纠错成功则将纠错后的数据提交，如果不成功，则发送第二次重传请求。发送端的第二次重传可以重传原信息码 I，也可以重传一次校验码 C_0，这取决于重传策略。

Ⅲ型 HEC 和Ⅱ型 HEC 的区别是，当数据发生错误时，接收端不保存错误数据，重传时发送端每次都会发送原始数据 I 和相应的检错纠错码。

5.6.4 信源信道联合编码

根据香农的分离原理：信源编码和信道编码可以分别设计，而且这种局部最优可使系统总体性能达到最优。但是这一重要结论的假设前提是：无论对于信源编码还是对于信道编码，需要假定可以容忍无限长的延时，即允许编码块无限长；必须预先掌握传输信道的统计特性。上述两条假设在实时的通信系统设计中往往得不到满足。

针对实际应用中的特殊性，人们提出了信源信道联合编码(joint source channel coding，JSCC)[70]技术。这是一种兼顾视频传输效率和质量的有效方法，目标是将信道带宽在信源和信道码率之间进行最优分配，使得端到端的失真达到最小，从而获得最佳的端到端传输性能。

根据码流的不同部分对于图像重建质量的作用不同，采用不同的保护机制，这是信源信道联合编码的一个主要应用。例如，视频包中的头信息是最重要的部分，对它采取的保护最多；运动矢量信息采取低一级的保护，最后是对纹理信息的保护。这样的保护机制能够在发生误码的情况下，尽可能地利用正确信息进行误码掩盖而不会使图像质量有过于严重的下降。

分层编码结合非对等错误保护，可以构成一种有效抗误码的信源/信道联合编码方案。在分级编码中，视频信号被分为两级以上的结构，基本层包含视频信号的基本信息，通过它可以恢复出一定质量的视频信号；增强层包含视频信号的细节信息，它可以提高视频的恢复质量。为了抵抗信道干扰，分层码流根据重要性程度的不同，采取不同的纠错码技术，保障基本层码流受损程度低，甚至不受损，从而保证接收端总可以得到一定质量的视频信号。

5.6.5 非对等保护

非对等保护(unequal error protection,UEP)[71]策略作为一种视频可靠传输的解决方案得到了广泛而深入的研究。其基本思想是,通过对重要的数据作更多的保护,使得重要的数据比特具有更强的错误恢复能力,从而达到整体解码最佳的效果。非对等保护策略主要有两个研究方向:

(1) 研究如何对信源编码端数据进行等级性划分,使码流适应不同网络状况的传输需要;

(2) 研究等级性划分后的数据采用怎样的不平等保护粒度问题,如调节 RS 码、Turbo 码等信道编码参数来实施不平等保护。

这两方面的研究都是为了使信源端编码后数据在经过复杂的信道环境影响后,接收端解码恢复的视频图像质量最佳。

5.6.6 差错隐藏

与数据传输不同,视频传输要求较高的实时性,能够容忍一定的传输错误。因此,视频传输错误无需完全恢复,可以使用一些相关数据来代替错误数据,从而实现误码隐藏(error concealment)。视频引用中,误码隐藏技术利用人眼的差错遮蔽特性以及视频信号的强相关性(如空间域和时间域)可以恢复出人眼可接受的视频信号。换句话说,误码隐藏并不真正消除误码,而是尽可能地弥补误码带来的视觉损伤。误码隐藏的前提是视频解码器必须首先检测出码流中是否存在误码,并且要尽可能精确地确定出误码位置。

(1) 错误检测。误码检测可在传输解码层和视频解码层进行。传输解码层可以通过在视频数据中插入检错/纠错码,如 BCH 码、RS 码、Turbo 码等,接收端可以检测到错误并确定错误的位置。在视频解码层,判断视频编码标准码流是否符合语法结构是视频解码层上检测误码的主要方法。这类方法利用视频信号本身属性及码流语法结构进行误码检测,不会增加额外的传输负担。当解码器发现这些规则遭到破坏时,就可以断定发生了误码。

(2) 错误隐藏。误码隐藏技术可分为时域误码掩盖和空域误码掩盖。时域误码掩盖是基于运动补偿的时间预测,它利用受损块的运动信息对图像进行恢复。为此,在 MPEG-2 标准中,即使是采用帧内编码模式的图像块,也允许传输相应的运动矢量,其目的就是为了有效恢复受损图像块。这种方法的局限性在于必须保证运动矢量的正确传输,运动矢量受损情况下就需要利用空域插值和时域插值的误码隐藏方法。空域误码掩盖方法有最优平滑恢复、凸集投影法、最小化相邻像素方差、临近像素插值等。

5.7 无线网络中多层非对等保护的动态优化组包策略

随着无线网络的快速发展和 Internet 中流媒体视频的巨大成功,无线网络中的视频服务有望在不久的将来得到大规模部署。但是,由于无线网络中有限的带宽和错误易发环境,

数据丢包和误码在无线流媒体业务中是不可避免的。为了提高无线流媒体业务的服务质量，研究人员设计了众多的差错控制算法。一个精心设计的组包策略不仅可以明显地促进流媒体视频抵抗误码能力，还能够减小压缩编码的负载。如何设计一个高效的抗错误的组包算法，是当前流媒体应用的一个备受关注的问题。本节针对无线视频应用，介绍一个多层非对等保护和动态优化的组包策略。

该策略同时考虑了无线网络传输中的丢包和误码两个问题，并尽可能地缓解它们所带来的画面失真。此外，该策略还提供了一个开放的框架，采用不同的工具就能适用于各种编码算法。多层非对等保护在应用层和链路层实现，目标在于降低丢包的概率。同时，内容相关的率失真优化算法可以最小化传输误码带来的画面失真，考虑了多层非对等保护带来的增益。模拟实验证明，与 MPEG-4 的组包策略相比，该方案在无线网络环境中能够显著地提升视频质量。

5.7.1　策略算法框架

由于当前的众多研究成果大多只考虑无线流媒体中的一个问题，传输丢包或者误码，或局限于某种编码算法中和某种编码结构中。部分研究集中在如何减轻丢包带来的视频画面失真。Wu 等人[72]研究了各种组包策略，并提出了一种针对多媒体比特流的全局优化和次优化的组包算法。Cai 等人提出了一种基于率失真 R-D(rate-distortion) 理论的优化组包策略，能够完全消除视频包间的相关性，适用于精细粒度可缩放（fine granularity scalability，FGS）比特流[73,74]，并引入非对等保护(unequal error protection，UEP）机制进一步提高了抗错误能力[75]。Chou 等人[76]针对通用可伸缩视频传输提出了相应的组包策略，并基于率失真理论提出了优化算法。一种源自适应的组包策略[77,78]使用了帧内包交织技术和前向纠错编码，试图降低无线网络中严重丢包带来的失真。

另一方面，一些研究者把注意力集中在错误易发网络中的传输误码上。Worrall[79]提出了一种运动-自适应的组包策略，通过动态改变视频包长度的算法达到优化的目的。另外，有人针对第三代移动网络 3G(third generation) 提出 MPEG-4 流媒体的优化组包策略，使用尽可能小的视频包长度并对视频包头进行压缩[80]。一种率失真优化的组包策略适用于错误易发传输通道中的 FGS 流[81]。

针对无线视频应用，我们提出了一个率失真优化的组包策略，它具有以下特点：

（1）属于信源信道联合编码，无需反馈，延时低，适用于 P2P 网络；

（2）是一个编码算法无关的策略框架，适用于各种视频编码算法；

（3）同时提供了针对传输丢包和误码的抗错误能力；

（4）采用了优先级分包策略和多层非对等保护(multi-layer unequal error protection，MUEP)，显著地增强了抗丢包能力；

（5）结合 MUEP 的影响，基于编码内容使用率失真理论进行优化，尽可能地减弱误码带来的图像失真。

本节提出的优化视频组包策略的根本出发点是同时考虑无线视频应用中的传输丢包和误码问题，从两个方面同时对组包算法进行最优化。主要包括动态优化分包和多层非对等保护等几个模块，其编码无关框架如图 5.7.1 所示。

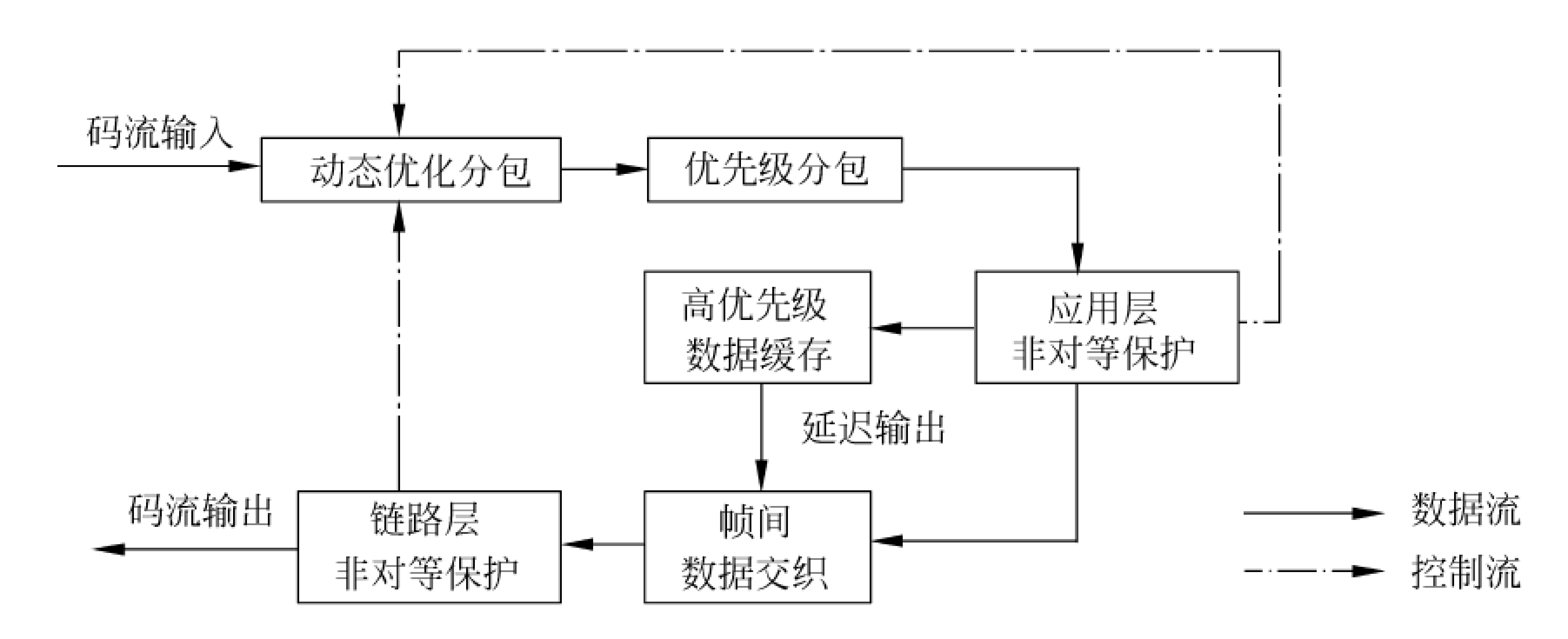


图 5.7.1　优化组包策略框架图

上述组包算法以一帧视频为组包对象，对编码后的视频流进行组包处理，工作流程如下：

（1）编码器对一帧视频进行编码，并记录各编码单元的内容信息，用于计算该单元可能的画面失真值，作为输入一起进入动态优化分包模块；

（2）动态优化分包模块结合相应的内容信息计算各编码单元的出错概率，此时需要综合考虑链路状态和后续多层非对等保护机制的增益，并利用动态规划算法计算出一个近似最优的分包结果，并输出分包后的子视频包；

（3）优先级分包模块对优化后的子视频包按照视频数据对解码过程的重要性，进一步进行分包：将每个子视频包分成一个高优先级和低优先级包；

（4）应用层非对等保护模块对高优先级包进行额外的处理，如采用前向纠错码或数据冗余方式等，生成相应的保护数据，送入高优先级数据缓存模块；为了避免无线网络中突发式的丢包或者误码，每份保护数据不与原始数据一起传输，而是强制延迟一定时间后与后续的原始数据交织在一起传输；延迟时间依赖于缓存区长度，同时应可根据网络状态动态地改变；

（5）帧间数据交织模块对子数据包重新组包，将相同优先级的子数据包组合成大的应用层数据包；对于高优先级数据，还将从数据缓存模块取出前面某帧的保护数据，与当前帧的原始数据交织编码，最后添加相应的流媒体传输协议头；

（6）链路层对不同优先级的数据作区分处理，高优先级数据享有更好的传输服务，如更高的重传次数上限，尽可能地保证高优先级数据的正确传输。

以 MPEG-4 视频编码算法为例，详细描述动态优化视频组包算法的具体流程。表 5.7.1 列出了其他部分使用的一些符号。

表 5.7.1　符号表

符号	描　　述	符号	描　　述
VOP_i	一个视频帧中的第 i 个视频对象平面	$D_{i,j,k}$	$\mathrm{MB}_{i,j,k}$ 损坏导致的画面失真
$\mathrm{VP}_{i,j}$	VOP_i 中的第 j 个视频包	$D^{i}_{i,j,k}$	$\mathrm{MB}_{i,j,k}$ 损坏经运动补偿后导致的画面失真
$\mathrm{MB}_{i,j,k}$	$\mathrm{VP}_{i,j}$ 中第 k 个宏块	P_e	传输误码率
$\mathrm{VP}^{h}_{i,j}$	$\mathrm{VP}_{i,j}$ 中的高优先级部分	$P^{h}_{i,j}$	$\mathrm{VP}_{i,j}$ 中高优先级部分的出错概率
$\mathrm{VP}^{l}_{i,j}$	$\mathrm{VP}_{i,j}$ 中的低优先级部分	$P^{l}_{i,j}$	$\mathrm{VP}_{i,j}$ 中低优先级部分的出错概率

在 MPEG-4 的一个视频帧中，其基本的编码单元是视频对象平面 VOP(visual object plane)，一个视频帧中包含了多个连续的 VOP，如图 5.7.2 所示。通过基于内容的率失真优化理论，每个 VOP 将被进一步分割成多个视频包 VP(video packet)，VOP_i 被分成 k 个 VP，即 $VP_{i,j}$，其中 $0 \leqslant j < k$ 。

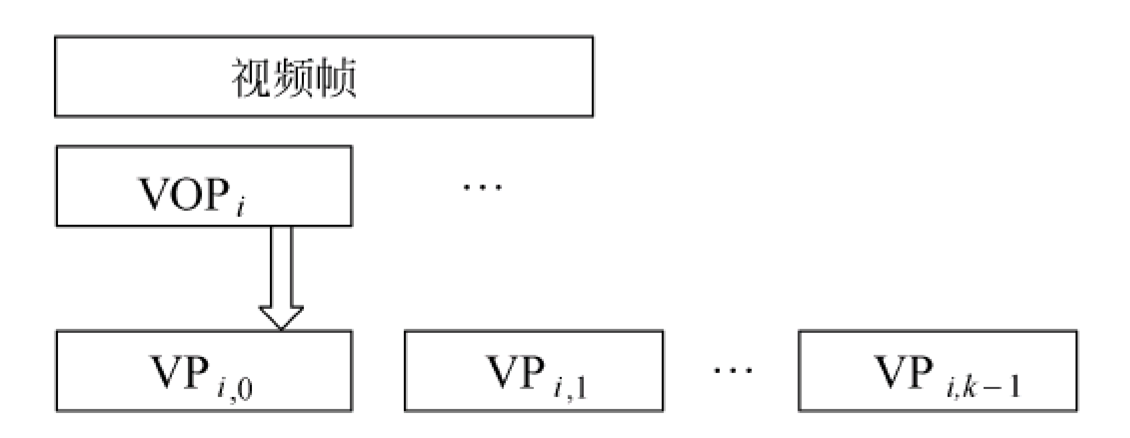


图 5.7.2　基于内容的率失真动态优化分包

视频包 VP 中的数据将根据在解码过程中重要性的不同被进一步分成两部分：高优先级部分和低优先级部分。如在 MPEG-4 编码标准中，视频包包头和运动向量数据对解码来说是必要的，该部分若损坏则整个视频包都将丢弃。而纹理数据即使丢失或者发生误码，也还能通过运动补偿恢复部分数据。因此，视频包头和运动数据组成新的高优先级包，而纹理数据则变为低优先级包。如图 5.7.3 所示，视频包 $VP_{i,j}$ 被进一步分成高优先级包 $VP^h_{i,j}$ 和低优先级包 $VP^l_{i,j}$ 。

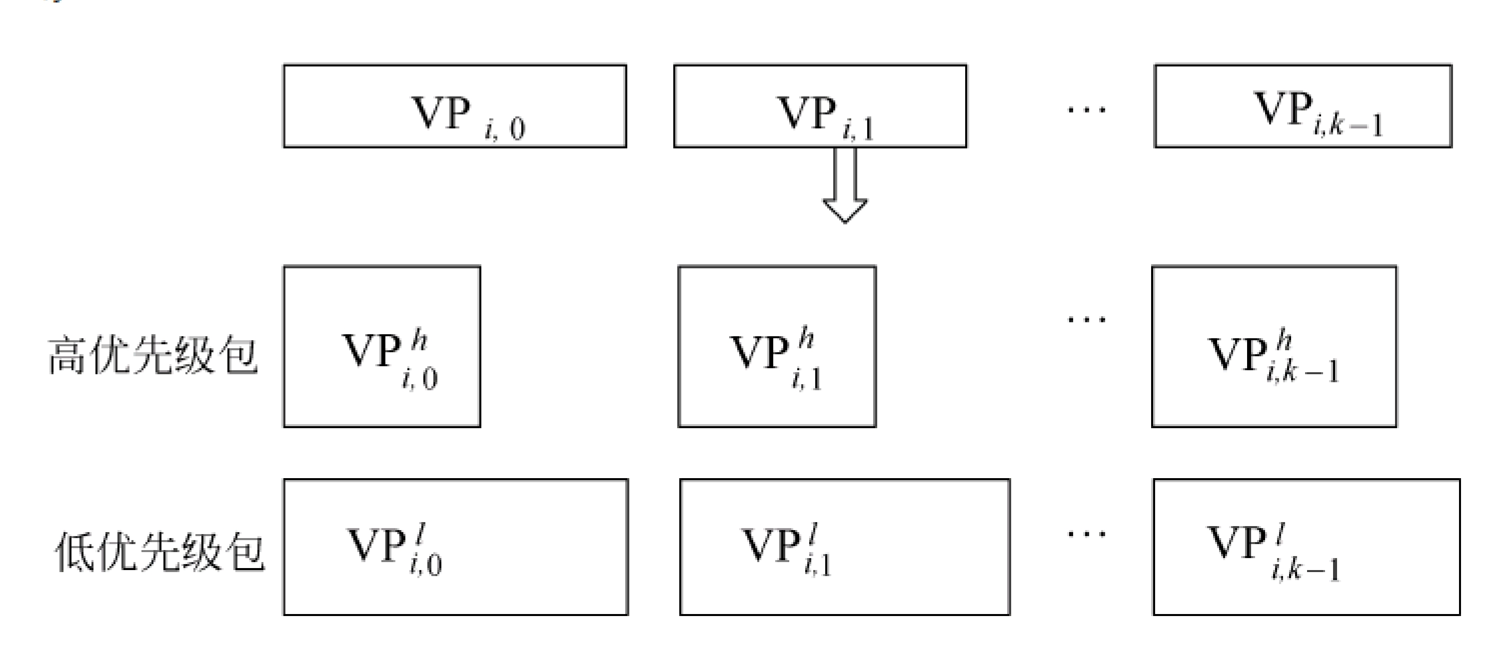


图 5.7.3　优先级分包

应用层非对等保护对 $VP^h_{i,j}$ 生成相应的保护数据，保护数据将延迟一定帧数后与某帧相应 VP 的原始数据。这种数据交织技术有利于分散传输错误的分布，提高数据恢复的可能性，如图 5.7.4 所示。

然后，这些包封装成实时传输协议 RTP(Real-time Transport Protocol)。为了减小包头的负荷和削弱 RTP 包间的依赖性，相同优先级的包将被填入同一个 RTP 包，直到没有足够空间为止。如图 5.7.5 所示，$m+1$ 个高优先级视频包 $VP^h_{i,j}(m<k, 0 \leqslant j \leqslant m)$ 封装成一个 RTP 包，$p+1$ 个低优先级视频包 $VP^l_{i,j}(p<k, 0 \leqslant j \leqslant p)$ 封装成另一个 RTP 包。RTP 包将通过 UDP-lite 和 IP 协议进一步封装。这里，UDP-lite 协议[82]可以用来将传输错误的包发送到应用层，而不是在传输层简单地丢弃。文献[83]对 IP/UDP-lite/RTP 协议头进行压缩，使用 IP 隧道机制[84]进行传输。

定理 5.7.1　若当前 RTP 包剩余空间能够完全容纳下一个视频包，则将该视频包填入当前的 RTP 包中。

定理 5.7.2　若当前 RTP 包剩余空间不能完全容纳下一个视频包，假设在用该视频包

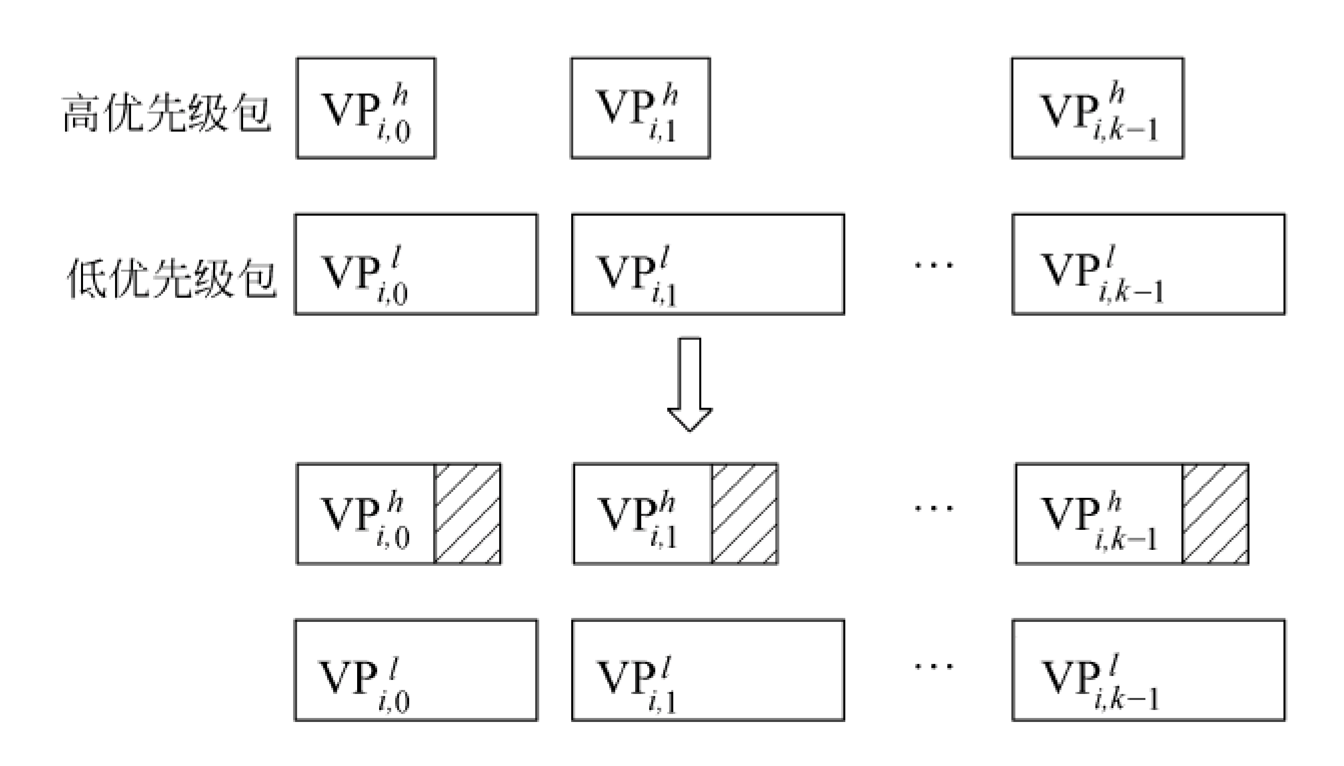


图 5.7.4　应用层非对等保护

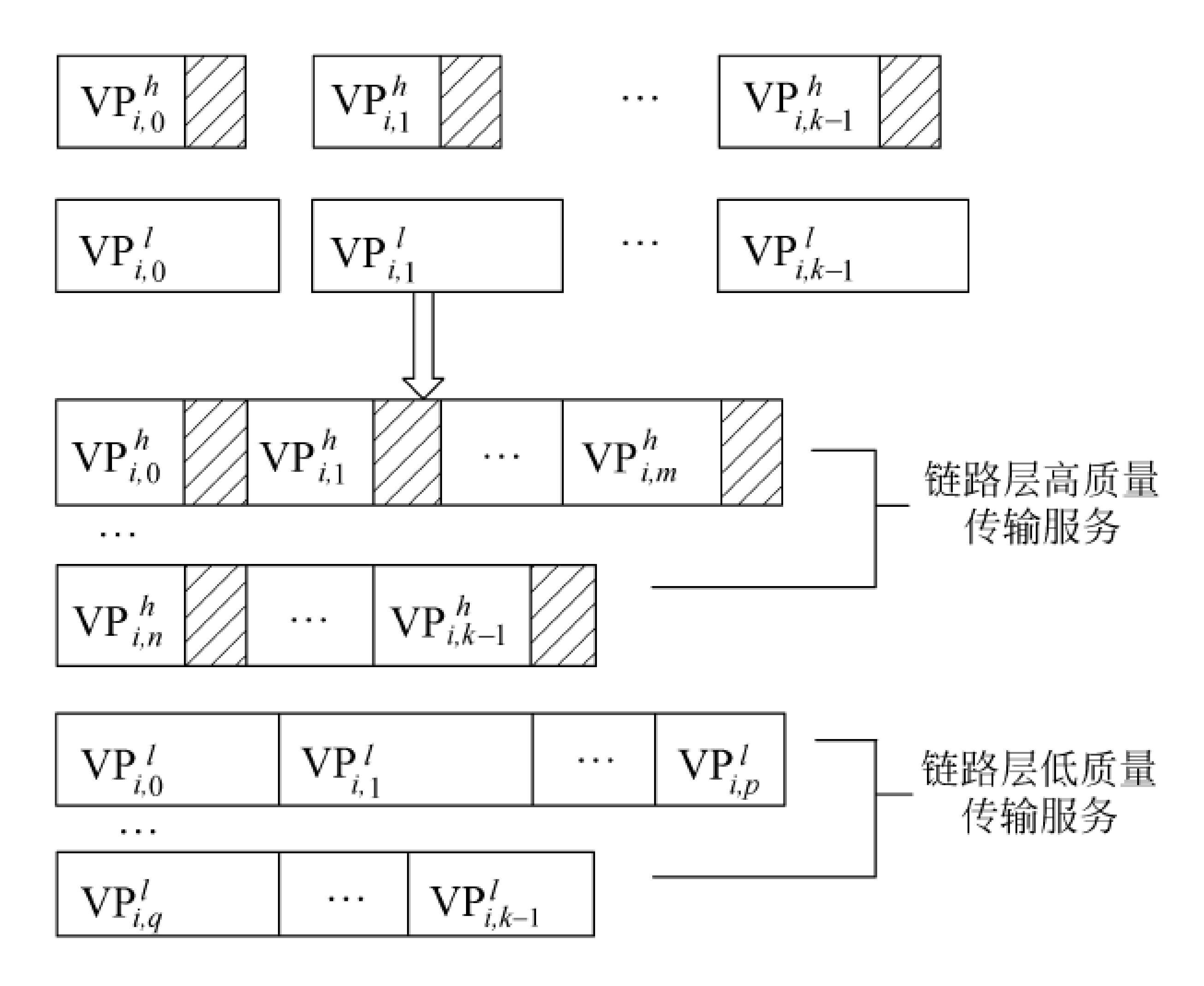


图 5.7.5　链路层非对等保护

填满当前 RTP 时该视频包将分布在 $m(m\geqslant 1)$个 RTP 包中，而重新开始一个 RTP 开始填入该视频包时，该视频包分步在 $n(n\geqslant 1)$个 RTP 包中，若满足 $m\leqslant n$，则将该视频包填入当前的 RTP 包中。

接收端收到数据包时，它首先检查视频数据帧的完整性和正确性。若发现高优先级视频包丢失或损坏，则缓存当前帧，直到相应的保护数据到达以后并对受损部分进行纠错；若纠错失败，则丢弃该视频包，可以使用错误隐藏机制。若低优先级数据被损坏，则忽略该低优先级视频包，并使用运动补偿技术进行解码。

5.7.2　动态优化算法

根据预先设定的参数，组包算法计算每个编码单元（包括高优先级视频包和低优先级视频包）的出错概率，并动态决定各个编码单元的组合方式，使得在概率统计的情况下该视频帧可能的画面失真最小。在 MPEG-4 中，我们需要动态决定每个视频包 VP（video packet）

中宏块MB(macro block)的数量,并固定一个视频帧中出现的VP的数量,以限制组包策略带来的负载。

假设一个视频帧中有L个MB和M个VOP。VOP_i包含N_i个VP,且$VP_{i,j}$包含了$N_{i,j}$个MB。为了限制视频包头带来的负荷,我们限制每个视频帧只能产生N个VP,则一个视频帧的概率平均失真可以由公式(5.7.1)表示:

$$D = \sum_{i=0}^{M-1} \sum_{j=0}^{N_i-1} \left(P_{i,j}^h \times \sum_{k=0}^{N_{i,j-1}} D_{i,j,k} + P_{i,j}^l \times \sum_{k=0}^{N_{i,j-1}} D'_{i,j,k} \right) \tag{5.7.1}$$

其中,

$$\sum_{i=0}^{M-1} N_i = N, \quad \sum_{i=0}^{M-1} \sum_{j=0}^{N_i} N_{i,j} = L \tag{5.7.2}$$

包的出错概率依赖于它的长度和信道的误码率。特别地,$VP_{i,j}^h$表示高优先级数据和它的保护数据同时出错的概率。假定高优先级包$VP_{i,j}^h$及其保护数据的长度分别为$L_{i,j}^h$和$L_{i,j}^{h,d}$,L_{hdr}是IP/UDP-lite/RTP包头的长度,$L_{i,j}^l$是低优先级包$VP_{i,j}^l$的长度,我们可以得到:

$$P_{i,j}^h = (1-(1-P_e)^{L_{\text{hdr}}+L_{i,j}^h}) \times (1-(1-P_e)^{L_{\text{hdr}}+L_{i,j}^{h,d}}) \tag{5.7.3}$$

$$P_{i,j}^l = (1-(1-P_e)^{L_{\text{hdr}}+L_{i,j}^h}) \tag{5.7.4}$$

另外,一个宏块导致的失真度可以由它的非零比特数来表示,这可以在编码过程中很容易获得,而且只需要消耗很小的额外负载。

上述优化问题的目标是找到一个$\{N_i^*, N_{i,j}^*: 0 \leqslant i < M, 0 \leqslant j < N_i^*\}$使得上述平均失真度$D$最小。为了提高该优化问题求解的效率,我们采用动态规划的方法进行求解,时间复杂度能够达到$O(N \times L)$。

5.7.3 多层对等保护

非对等保护在应用层和链路层实现。在无线网络中的链路层,非对等保护可以通过采用不同的重传次数上限来实现。对于高优先级的IP/UDP-lite/RTP包使用更高的重传次数上限,而低优先级包使用更高的重传次数上限,使得高优先级数据能够以更大的成功概率传输到目的地。

在应用层,针对不同的编码算法可以选用不同的工具。信道纠错编码是内容无关的纠错算法,适用于所有的编码算法。特别地,文献[85]提出了一种适合MPEG-4编码标准的轻型数据冗余算法。它只对每个宏块中最重要的少量数据进行冗余,以较小的代价实现高效的高优先级数据保护。

5.7.4 组包算法评价

我们在MPEG-4编码基础上实现了一个组包策略的系统原型,对采用模拟工具NS-2及相应的扩展插件[86]进行性能评价。这里使用两个QCIF(quarter common intermediate format)标准测试序列:Foreman和Suzie。两个测试序列分别在MPEG-4标准编码器和我们的系统原型上进行编码。编码过程基于以下假定:

(1) 测试序列的第一帧为帧内编码(I帧),其余使用帧间编码(P帧);

(2) I帧总是可以正确地传输。

解码端采用了简单的错误隐藏机制。若高优先级数据被损坏，相应的视频包将被丢弃，并使用前一帧相同位置的图像来替代。若低优先级数据即纹理数据被损坏，该视频包将使用运动补偿的方式进行解码。性能评价主要比较我们的组包策略和 MPEG-4 标准的组包策略，同时考虑了丢包率(packet loss rate，PLR)和误码率(bit error rate，BER)。下面的每个模拟都进行 30 次并取平均结果。

采用峰值信噪比(peak signal noise ratio，PSNR)作为解码端重构图像的客观标准。PSNR 的计算公式如式(5.7.5)和式(5.7.6)所示：

$$\mathrm{PSNR} = 10\lg\frac{255^2}{\mathrm{MSE}} \tag{5.7.5}$$

$$\mathrm{MSE} = \frac{\sum_{i=0}^{w}\sum_{j=0}^{h}(f(i,j)' - f(i,j))^2}{w \times h} \tag{5.7.6}$$

其中，w 和 h 分别为图像的宽和高，$f(i,j)$是像素点(i,j)的原始值，$f(i,j)'$是该像素点解码后的值，MSE 为均方误差。

1. 不同丢包率下的性能评价

当误码率 BER 为 10^{-4}时，模拟了不同丢包率 PLR 下两种组包策略的性能，分别为 0，5%，10%，20%。两个测试序列的相应结果如图 5.7.6 和图 5.7.7 所示。当没有丢包即丢包率为 0 时，相对于传统方案，我们的方案分别获得了 2.66 dB 和 2.72 dB 的增益。可见率失真优化有效地降低了误码带来的画面失真。另外，应用层的非对等保护同样改善了误码带来的失真。当丢包率增加时，两种组包策略的 PSNR 值都有所下降，但方案的下降速率远远低于传统的方案。如图所示，当丢包率为 20%时，方案分别能获得 4.22 dB(Foreman)和 3.05 dB(Suzie)的增益。

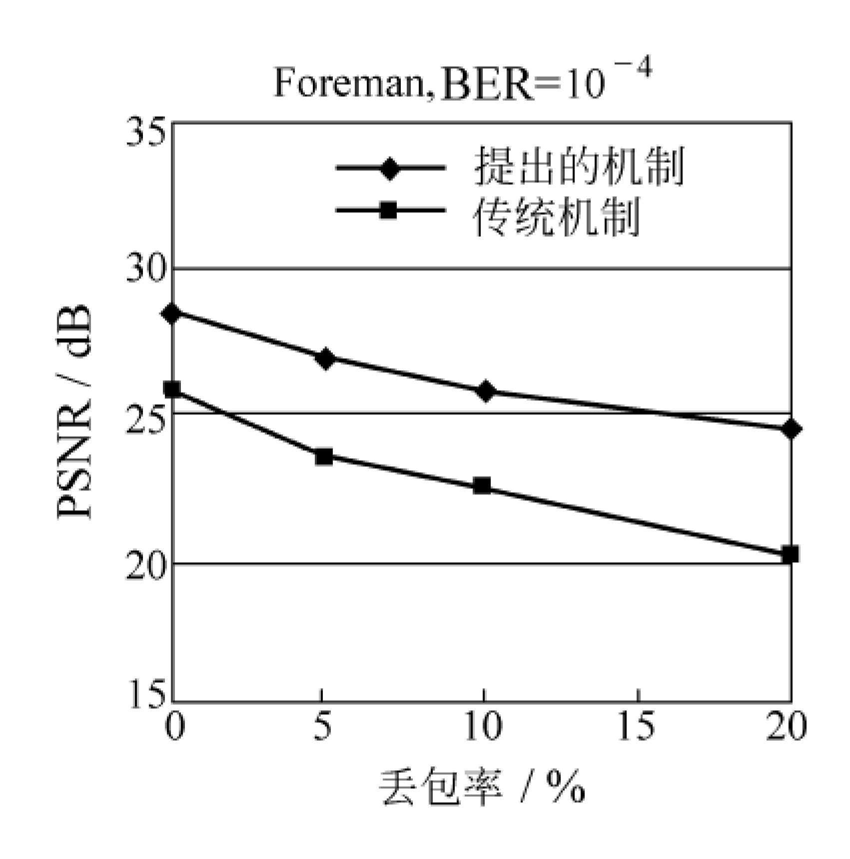


图 5.7.6 不同丢包率下 Foreman 序列性能比较

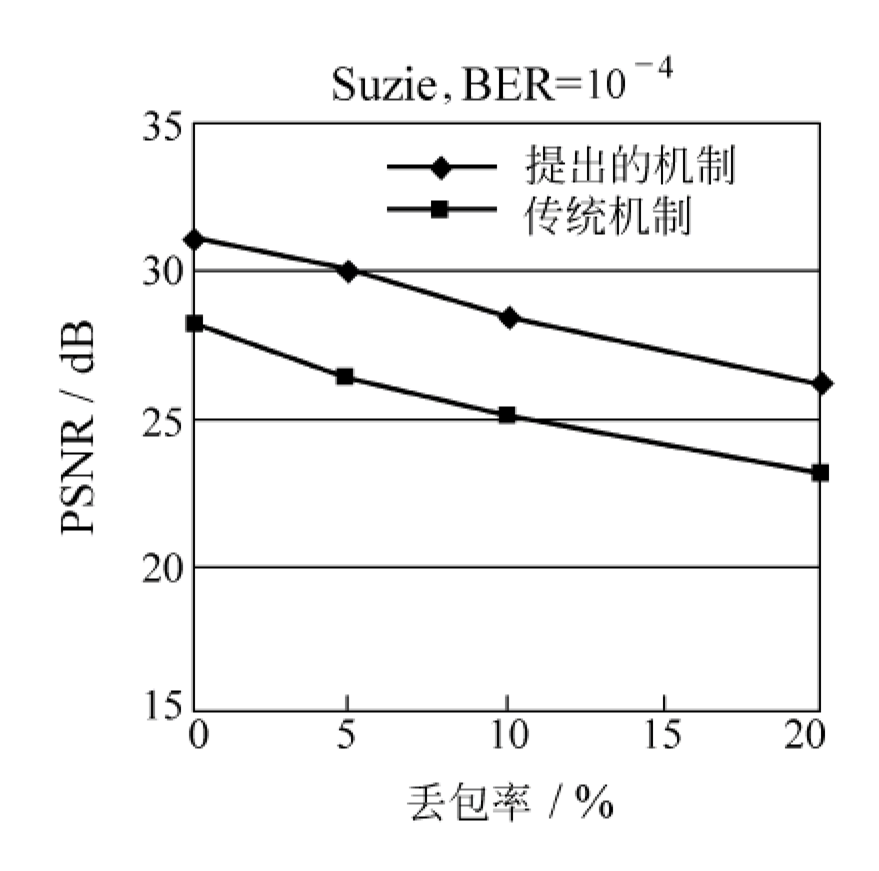


图 5.7.7 不同丢包率下 Suzie 序列性能比较

2. 不同误码率下的性能评价

当丢包率为 5%时，比较在不同的误码率下的性能，误码率分别为 0，10^{-4}，4×10^{-4}和 10^{-3}。图 5.7.8 和图 5.7.9 分别描述了两个测试序列的模拟结果。由图可以看出，当没有误码时，本节方案的 PSNR 远远超过了传统方案。这是因为多层的非对等保护有效地减少

了高优先级数据的丢失概率。误码率增加显著加剧了高优先级数据和低优先级数据的受损概率，导致视频质量剧烈下降。当误码率为 10^{-3} 时，本节提出的方案比传统的方案获得了 2.26 dB 和 2.17 dB 的增益。然而由于组包策略缺乏网络探测机制，不能自适应网络状况，这可以作为下一步的研究工作。

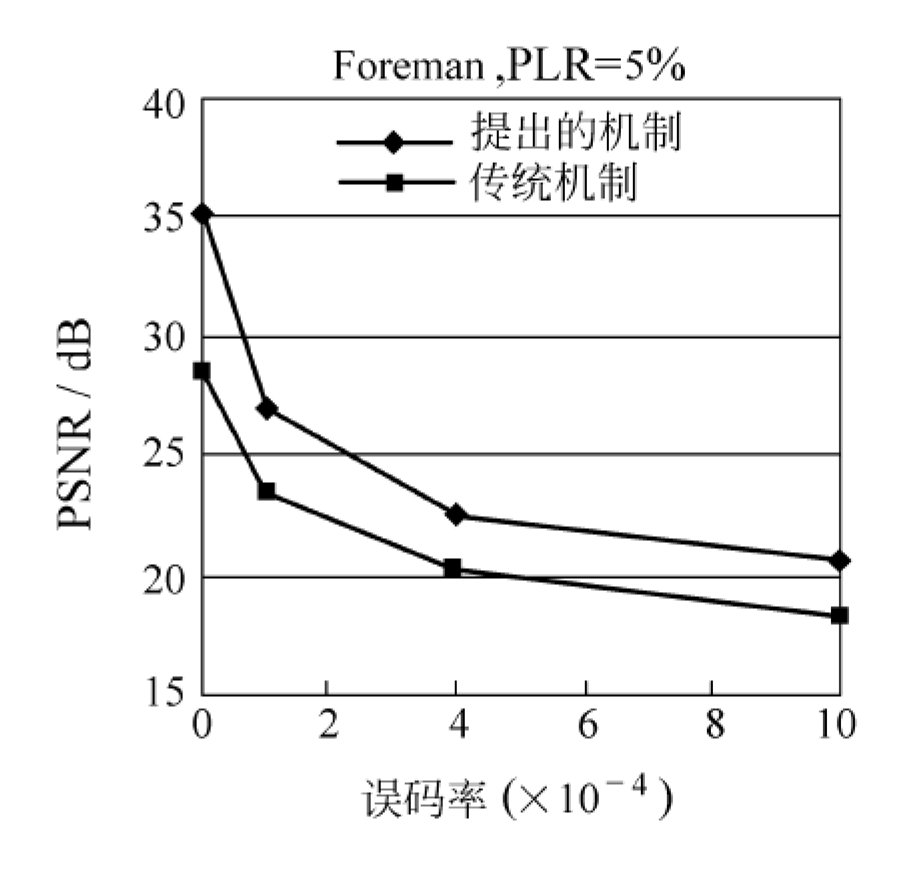


图 5.7.8　不同误码率下 Foreman 序列性能比较

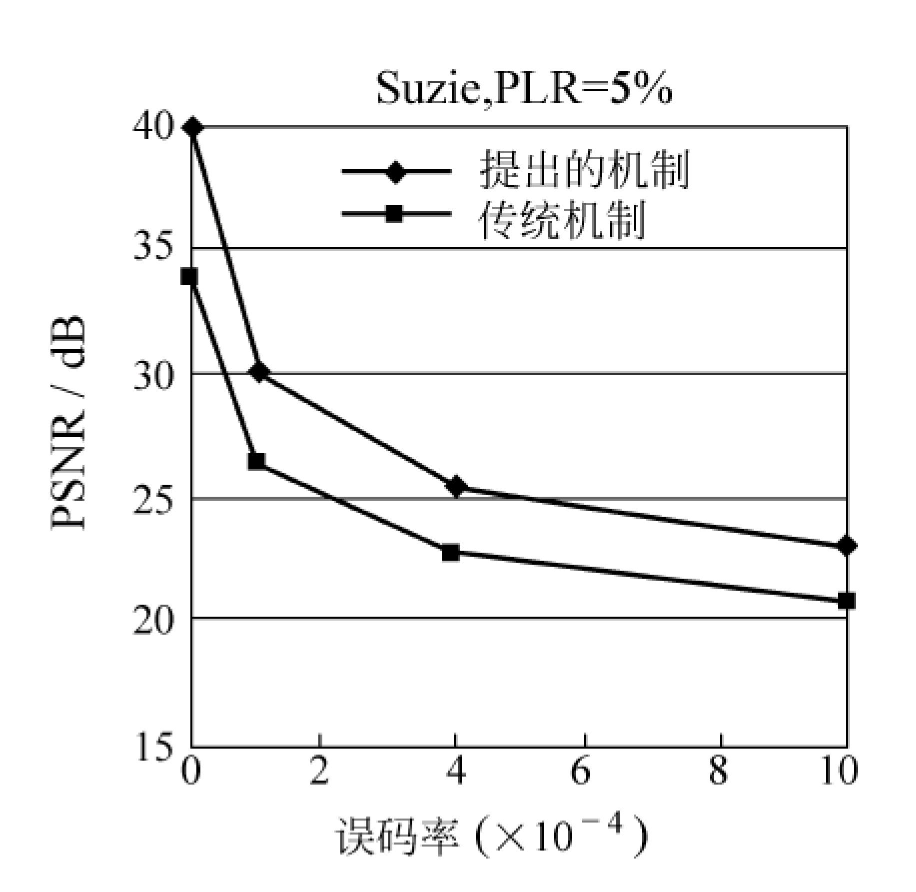


图 5.7.9　不同误码率下 Suzie 序列的性能比较

参 考 文 献

1　Suman Banerjee，Bobby Bhatacharjee，Christopher Kommareddy. Scalable Application Layer Multicast. ACM SIGCOMM，2002，43～51

2　Chu Y，Rao S，Seshan S，et al. Enabling conferencing applications on the Internet using an overlay multicast architecture. ACM SIGCOMM，San Diego，August 2001

3　Zhang B，Jamin S，Zhang L. Host multicast：A frame work for delivering multicast to end users. In：Proceedings of IEEE INFOCOM，June 2002

4　Banerjee S，Kommareddy C，Kar K，et al. Construction of an efficient overlay multicast infrastructure for real-time applications. IEEE INFOCOM，San Francisco，April，2003

5　Deshpande H，Bawa M，Garcia Molina H. Streaming live media over a peer-to-peer network. Technical Report，Stanford University，April 2001

6　Padmanabhan V，Wang H，Chou P，et al. Distributing streaming media content using cooperative networking. ACM NOSSDAV，Miami，May 2002

7　Castro M，D ruschel P，Kermarrec A M，et al. SplitStream：High-bandwidth content distribution in a cooperative environment. IPTPS'03，Berkeley，Feb. 2003

8　Wade Trappe，Jie Song，Radha Poovendran，Liu K J Ray. Key management and distribution for secure multimedia multicast. IEEE Trans on Multimedia，2003，5(4)

9　Chu Hao-Hua，Qiao Lintian，Klara Nahrstedt. A secure multicast protocol with copyright protection. ACM SIGCOMM Computer Communications Review，2002，32(2)

10　尹浩，林闯，邱锋，丁嵘. 数字水印技术综述. 计算机研究与发展，2005，42(7)：1093～1099

11　章淼，徐明伟，吴建平. 应用层组播研究综述. 电子学报，2004

12　Polk W T，Dodson D F，et al. Public key infrastructure：From theory to implementation. http://csrc.

ncsl. nist. gov/pki/panel/overview. html，NIST
13 Wade Trappe，Jie Song，Radha Poovendran，Liu K J Ray. Key distribution for secure multimedia multicast via data embedding. IEEE 2001
14 Poovendran R，Baras J S. An information-theoretic approach for design and analysis of rooted-tree-based multicast key management schemes. IEEE Trans Information Theory，2001，47：2824～2834
15 Voyatzis G，Pitas I. The use of watermarks in the protection of digital multimedia products. In：Proceedings of the IEEE，1999，87(7)：1197～1207
16 Cox I J，Linnartz J P. Some general methods for tampering with watermarks. IEEE J Selected Areas Communication，1998，16：587～593
17 Bender W，Gruhl D，Morimoto N，et al. Techniques for data hiding. IBM System Journal，1996
18 Wolfgang R B，Delp E J. A watermark for still images. In：Proc Int'l Conf on Image Processing，1996
19 Tirkel A Z，Osborne C F，Hall T E. Image and watermark registration. Signal Processing，1998，66(3)：373～383
20 Cox I J，Kilian J，Leighton F T，Shamoon T. Secure spread spectrum watermarking for multimedia. IEEE Trans on Image Processing，1997，6(12)：1673～1687
21 Lee Sin-Joo，Jung Sung-Hwan. A survey of watermarking techniques applied to multimedia. In：Proceedings IEEE International Symposium on Industrial Electronics (ISIE 2001)，2001，1：272～277
22 Podilchuk C I，Delp E J. Digital watermarking：Algorithms and applications. IEEE Signal Processing Magazine，2001，18(4)：33～46
23 Voloshynovskiy S，Pereira S，Pun T，et al. Attacks on digital watermarks：Classification，estimation-based attacks and benchmarks. IEEE Communications Magazine，2001，39(8)：118～126
24 Podilchuk C I，Zeng W. Perceptual watermarking of still images. In：Proc IEEE Signal Processing Society 1997，Workshop Signal Processing，June 23—25，1997，363～368
25 Yu Nenghai，Cao liangliang，Fang Wen，Li Xuelong. Practical analysis of watermarking capacity. In：Proc of the International Conference on Communication Technology (ICCT 2003)，2003，2：9～11
26 Liu Tong，Qiu Zhengding. Attacks and evaluation in image digital watermarking. Information and Control，2001，30(5)
27 Craver S，Memon N，Yeo B L，et al. On the invertibility of invisible watermarking techniques. ICIP-1997：540～543
28 Craver S，Memon N，et al. Can invisible watermarks resolve rightful ownerships? IBM Research Tech. Rep，Rc20509，IBM Cyber Journal，1997
29 Mukherjee D P，Maitra S，Acton S T. Spatial domain digital watermarking of multimedia objects for buyer authentication. IEEE Trans on Multimedia，2004，6(1)：1～15
30 van Schyndel R G，Tirkel A Z，Osborne C F. A digital watermark. In：Proceedings IEEE International Conference Image Processing (ICIP-94)，1994，2
31 Hsu C T，Wu J L. Hidden digital watermarks in images. IEEE Trans on Image Processing，1999，8(1)：58～68
32 Wolfgang R B，Delp E J. A watermark for digital images. In：Proc ICIP'96，Lausanne，Switzerland，Sept. 1996
33 Kutter M，Jordan F，Bosson F. Digital signature of color images using amplitude modulation. In：Proc of the SPIE，1997，3022：518～526
34 Pitas. A method for signature casting on digital image. In：Proc of ICIP，1996，3：215～218
35 Hartung F，Girod B. Digital watermarking of MPEG-2 coded video in the bitstream domain. In：Proc

ncsl. nist. gov/pki/panel/overview. html,NIST
13 Wade Trappe,Jie Song, Radha Poovendran, Liu K J Ray. Key distribution for secure multimedia multicast via data embedding. IEEE 2001
14 Poovendran R, Baras J S. An information-theoretic approach for design and analysis of rooted-tree-based multicast key management schemes. IEEE Trans Information Theory,2001,47: 2824～2834
15 Voyatzis G, Pitas I. The use of watermarks in the protection of digital multimedia products. In: Proceedings of the IEEE,1999,87(7): 1197～1207
16 Cox I J,Linnartz J P. Some general methods for tampering with watermarks. IEEE J Selected Areas Communication,1998,16: 587～593
17 Bender W,Gruhl D,Morimoto N,et al. Techniques for data hiding. IBM System Journal,1996
18 Wolfgang R B,Delp E J. A watermark for still images. In: Proc Int'l Conf on Image Processing,1996
19 Tirkel A Z,Osborne C F,Hall T E. Image and watermark registration. Signal Processing,1998,66(3): 373～383
20 Cox I J,Kilian J,Leighton F T,Shamoon T. Secure spread spectrum watermarking for multimedia. IEEE Trans on Image Processing,1997,6(12): 1673～1687
21 Lee Sin-Joo,Jung Sung-Hwan. A survey of watermarking techniques applied to multimedia. In: Proceedings IEEE International Symposium on Industrial Electronics (ISIE 2001),2001,1: 272～277
22 Podilchuk C I,Delp E J. Digital watermarking: Algorithms and applications. IEEE Signal Processing Magazine,2001,18(4): 33～46
23 Voloshynovskiy S,Pereira S,Pun T,et al. Attacks on digital watermarks: Classification,estimation-based attacks and benchmarks. IEEE Communications Magazine,2001,39(8): 118～126
24 Podilchuk C I,Zeng W. Perceptual watermarking of still images. In: Proc IEEE Signal Processing Society 1997,Workshop Signal Processing,June 23—25,1997,363～368
25 Yu Nenghai,Cao liangliang,Fang Wen,Li Xuelong. Practical analysis of watermarking capacity. In: Proc of the International Conference on Communication Technology (ICCT 2003),2003,2: 9～11
26 Liu Tong,Qiu Zhengding. Attacks and evaluation in image digital watermarking. Information and Control,2001,30(5)
27 Craver S,Memon N,Yeo B L,et al. On the invertibility of invisible watermarking techniques. ICIP-1997: 540～543
28 Craver S,Memon N,et al. Can invisible watermarks resolve rightful ownerships? IBM Research Tech. Rep,Rc20509,IBM Cyber Journal,1997
29 Mukherjee D P,Maitra S, Acton S T. Spatial domain digital watermarking of multimedia objects for buyer authentication. IEEE Trans on Multimedia,2004,6(1): 1～15
30 van Schyndel R G,Tirkel A Z,Osborne C F. A digital watermark. In: Proceedings IEEE International Conference Image Processing (ICIP-94),1994,2
31 Hsu C T,Wu J L. Hidden digital watermarks in images. IEEE Trans on Image Processing,1999,8(1): 58～68
32 Wolfgang R B,Delp E J. A watermark for digital images. In: Proc ICIP'96,Lausanne,Switzerland, Sept. 1996
33 Kutter M,Jordan F,Bosson F. Digital signature of color images using amplitude modulation. In: Proc of the SPIE,1997,3022: 518～526
34 Pitas. A method for signature casting on digital image. In: Proc of ICIP,1996,3: 215～218
35 Hartung F,Girod B. Digital watermarking of MPEG-2 coded video in the bitstream domain. In: Proc

ncsl. nist. gov/pki/panel/overview. html，NIST
13 Wade Trappe，Jie Song，Radha Poovendran，Liu K J Ray. Key distribution for secure multimedia multicast via data embedding. IEEE 2001
14 Poovendran R，Baras J S. An information-theoretic approach for design and analysis of rooted-tree-based multicast key management schemes. IEEE Trans Information Theory，2001，47：2824～2834
15 Voyatzis G，Pitas I. The use of watermarks in the protection of digital multimedia products. In：Proceedings of the IEEE，1999，87(7)：1197～1207
16 Cox I J，Linnartz J P. Some general methods for tampering with watermarks. IEEE J Selected Areas Communication，1998，16：587～593
17 Bender W，Gruhl D，Morimoto N，et al. Techniques for data hiding. IBM System Journal，1996
18 Wolfgang R B，Delp E J. A watermark for still images. In：Proc Int'l Conf on Image Processing，1996
19 Tirkel A Z，Osborne C F，Hall T E. Image and watermark registration. Signal Processing，1998，66(3)：373～383
20 Cox I J，Kilian J，Leighton F T，Shamoon T. Secure spread spectrum watermarking for multimedia. IEEE Trans on Image Processing，1997，6(12)：1673～1687
21 Lee Sin-Joo，Jung Sung-Hwan. A survey of watermarking techniques applied to multimedia. In：Proceedings IEEE International Symposium on Industrial Electronics (ISIE 2001)，2001，1：272～277
22 Podilchuk C I，Delp E J. Digital watermarking：Algorithms and applications. IEEE Signal Processing Magazine，2001，18(4)：33～46
23 Voloshynovskiy S，Pereira S，Pun T，et al. Attacks on digital watermarks：Classification，estimation-based attacks and benchmarks. IEEE Communications Magazine，2001，39(8)：118～126
24 Podilchuk C I，Zeng W. Perceptual watermarking of still images. In：Proc IEEE Signal Processing Society 1997，Workshop Signal Processing，June 23—25，1997，363～368
25 Yu Nenghai，Cao liangliang，Fang Wen，Li Xuelong. Practical analysis of watermarking capacity. In：Proc of the International Conference on Communication Technology (ICCT 2003)，2003，2：9～11
26 Liu Tong，Qiu Zhengding. Attacks and evaluation in image digital watermarking. Information and Control，2001，30(5)
27 Craver S，Memon N，Yeo B L，et al. On the invertibility of invisible watermarking techniques. ICIP-1997：540～543
28 Craver S，Memon N，et al. Can invisible watermarks resolve rightful ownerships? IBM Research Tech. Rep，Rc20509，IBM Cyber Journal，1997
29 Mukherjee D P，Maitra S，Acton S T. Spatial domain digital watermarking of multimedia objects for buyer authentication. IEEE Trans on Multimedia，2004，6(1)：1～15
30 van Schyndel R G，Tirkel A Z，Osborne C F. A digital watermark. In：Proceedings IEEE International Conference Image Processing (ICIP-94)，1994，2
31 Hsu C T，Wu J L. Hidden digital watermarks in images. IEEE Trans on Image Processing，1999，8(1)：58～68
32 Wolfgang R B，Delp E J. A watermark for digital images. In：Proc ICIP'96，Lausanne，Switzerland，Sept. 1996
33 Kutter M，Jordan F，Bosson F. Digital signature of color images using amplitude modulation. In：Proc of the SPIE，1997，3022：518～526
34 Pitas. A method for signature casting on digital image. In：Proc of ICIP，1996，3：215～218
35 Hartung F，Girod B. Digital watermarking of MPEG-2 coded video in the bitstream domain. In：Proc

ncsl. nist. gov/pki/panel/overview. html, NIST
13 Wade Trappe, Jie Song, Radha Poovendran, Liu K J Ray. Key distribution for secure multimedia multicast via data embedding. IEEE 2001
14 Poovendran R, Baras J S. An information-theoretic approach for design and analysis of rooted-tree-based multicast key management schemes. IEEE Trans Information Theory, 2001, 47: 2824～2834
15 Voyatzis G, Pitas I. The use of watermarks in the protection of digital multimedia products. In: Proceedings of the IEEE, 1999, 87(7): 1197～1207
16 Cox I J, Linnartz J P. Some general methods for tampering with watermarks. IEEE J Selected Areas Communication, 1998, 16: 587～593
17 Bender W, Gruhl D, Morimoto N, et al. Techniques for data hiding. IBM System Journal, 1996
18 Wolfgang R B, Delp E J. A watermark for still images. In: Proc Int'l Conf on Image Processing, 1996
19 Tirkel A Z, Osborne C F, Hall T E. Image and watermark registration. Signal Processing, 1998, 66(3): 373～383
20 Cox I J, Kilian J, Leighton F T, Shamoon T. Secure spread spectrum watermarking for multimedia. IEEE Trans on Image Processing, 1997, 6(12): 1673～1687
21 Lee Sin-Joo, Jung Sung-Hwan. A survey of watermarking techniques applied to multimedia. In: Proceedings IEEE International Symposium on Industrial Electronics (ISIE 2001), 2001, 1: 272～277
22 Podilchuk C I, Delp E J. Digital watermarking: Algorithms and applications. IEEE Signal Processing Magazine, 2001, 18(4): 33～46
23 Voloshynovskiy S, Pereira S, Pun T, et al. Attacks on digital watermarks: Classification, estimation-based attacks and benchmarks. IEEE Communications Magazine, 2001, 39(8): 118～126
24 Podilchuk C I, Zeng W. Perceptual watermarking of still images. In: Proc IEEE Signal Processing Society 1997, Workshop Signal Processing, June 23—25, 1997, 363～368
25 Yu Nenghai, Cao liangliang, Fang Wen, Li Xuelong. Practical analysis of watermarking capacity. In: Proc of the International Conference on Communication Technology (ICCT 2003), 2003, 2: 9～11
26 Liu Tong, Qiu Zhengding. Attacks and evaluation in image digital watermarking. Information and Control, 2001, 30(5)
27 Craver S, Memon N, Yeo B L, et al. On the invertibility of invisible watermarking techniques. ICIP-1997: 540～543
28 Craver S, Memon N, et al. Can invisible watermarks resolve rightful ownerships? IBM Research Tech. Rep, Rc20509, IBM Cyber Journal, 1997
29 Mukherjee D P, Maitra S, Acton S T. Spatial domain digital watermarking of multimedia objects for buyer authentication. IEEE Trans on Multimedia, 2004, 6(1): 1～15
30 van Schyndel R G, Tirkel A Z, Osborne C F. A digital watermark. In: Proceedings IEEE International Conference Image Processing (ICIP-94), 1994, 2
31 Hsu C T, Wu J L. Hidden digital watermarks in images. IEEE Trans on Image Processing, 1999, 8(1): 58～68
32 Wolfgang R B, Delp E J. A watermark for digital images. In: Proc ICIP'96, Lausanne, Switzerland, Sept. 1996
33 Kutter M, Jordan F, Bosson F. Digital signature of color images using amplitude modulation. In: Proc of the SPIE, 1997, 3022: 518～526
34 Pitas. A method for signature casting on digital image. In: Proc of ICIP, 1996, 3: 215～218
35 Hartung F, Girod B. Digital watermarking of MPEG-2 coded video in the bitstream domain. In: Proc

IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-97), 1997, Vol. 4
36 Hartung F, Girod B. Digital watermarking of uncompressed and compressed video. Signal Processing (Special Issue on Copyright Protection and Access Control for Multimedia Services), 1998, 66(3): 283～301
37 Langelaar G C, Lagendijk R L, Biemond J. Real-time labeling methods for MPEG compressed video. In: Proc of the 18th Symp Information Theroy in the Benelus, 1997
38 Jordan F, Kutter M, Ebrahimi T. Proposal of a watermarking technique for hiding/retrieving data in compressed and decompressed video. ISO/IEC Doc. JTC1/SC29/WG11 MPEG97/M2281, 1997
39 Cheng Hui, Isnardi M A. Spatial temporal and histogram video registration for digital watermark detection. In: Proceedings International Conference on Image Processing, 2003, Vol. 2
40 Busch C, Funk W, Wolthusen S. Digital watermarking: From concepts to real-time video applications. IEEE Computer Graphics and Applications, 1999, 19(1)
41 Hsieh Ming-Shing, Tseng Din-Chang, Huang Yong-Huai. Hiding digital watermarks using multiresolution wavelet transform. IEEE Trans on Industrial Electronics, 2001, 48(5)
42 Guo Huiping, Georganas N D. Multi-resolution image watermarking scheme in the spectrum domain. In: IEEE Canadian Conference on Electrical and Computer Engineering (CCECE 2002), 2002, Vol. 2: 12～15
43 Dugelay J L, Roche S. Fractal transform based large digital watermark embedding and robust full blind extraction. In: Proc IEEE International Conference on Multimedia Computing and Systems, 1999, Vol. 2
44 Bas P, Chassery J-M, Davoine F. Using the fractal code to watermark images. In: Proc IEEE International Conf on Image Processing (ICIP-98), 1998, 1: 469～473
45 Jacquin A E. Image coding based on a fractal theory of iterated contractive image transformations. IEEE Trans on Image Processing, 1992, 1(1): 18～30
46 Li Y, Chen Z, Tan S, Campbell R. Security enhanced MPEG player. In: Proceedings of IEEE First International Workshop on Multimedia Software Development (MMSD'96), Berlin, Germany, March 1996
47 Maples T B, Spanos G A. Performance study of a selective encryption scheme for the security of networked, real-time video. In: Proceedings of the 4th International Conference on Computer Communication and Network, Las Vegas, Navada, September 1995
48 Agi I, Gong L. An empirical study of MPEG video transmission. In: Proceedings of the Internet Society Symposium on Network and Distributed System Security, San Diego, CA, February 1996
49 Meyer J, Gadegast F. Security mechanisms for multimedia data with the example of MPEG-1 video. http://www.powerweb.de/phade/phade.html
50 Tang L. Methods for encrypting and decrypting MPEG video data efficiently. In: Proceedings of the Fourth ACM International Multimedia Conference (ACM Multimedia'96), Boston, MA, Nov. 1996
51 Qiao L, Nahrstedt K. Comparison of MPEG encryption algorithms. International Journal on Computers and Graphics (Special Issue: Data Security in Image Communication and Network), 1998, 22(3)
52 Qiao L, Nahrstedt K. A new algorithm for MPEG video encryption. In: Proceedings of the First International Conference on Imaging Science, Systems and Technology (CISST'97), Las Vegas, Nevada, July 1997, 21～29
53 Shi C, Bhargava B. A fast MPEG video encryption algorithm. In: Proceedings of the 6th ACM

International Multimedia Conference, Bristol, UK, Sept. 1998
54 Shi C, Bhargava B. An efficient MPEG video encryption algorithm. In: Proceedings of the 17th IEEE Symposium on Reliable Distributed Systems, West Lafayette, Indiana, Oct. 1998
55 Podesser M, Schmidt H-P, Uhl A. Selective bitplane encryption for secure transmission of image data in mobile enviroment. In: Proceedings of the 5th IEEE Nordic Signal Processing Symposium (NORSIG 2002), Tromso-Trondheim, Norway, October 2002
56 Sandro Rafaeli, David Hutchison. A survey of key management for secure group communication. ACM Computing Surveys, 2003, 35(3): 309～329
57 Rafaeli S. A decentralized architecture for group key management. PhD appraisal, Lancaster University, Lancaster, UK, September 2000
58 Perrig A, Song D, Tygar J D. ELK: A new protocol for efficient large-group key distribution. In: Proceedings of the IEEE Symposium on Security and Privacy, Los Alamitos, CA: IEEE Computer Society Press, 2001
59 Solomon G. Self-synchronizing Reed-Solomon codes (Corresp.). IEEE Trans on Information Theory, 1968, 14(4): 608～609
60 张春田，苏育挺，张静. 数字图像压缩编码. 北京：清华大学出版社，2004
61 MPEG-4 Video Verification Model version 18.0. Technical report, January, 2001
62 MPEG-4 Overview (V.21-Jeju Version). ISO/IEC JTC1/SC29/WG11 N4668, 2002
63 钟玉琢，王琪，贺玉文. 基于对象的多媒体数据压缩编码国际标准——MPEG-4 及其校验模型. 北京：科学出版社，2000
64 Wiegand T, Sullivan G J. Overview of the H.264/AVC video coding standard. IEEE Trans on Circuit System Video Technology, 2003, 13: 560～576
65 Chung D, Wang Y. Multiple description image coding using signal decomposition and reconstruction based on lapped orthogonal transforms. IEEE Trans Circuit and System for Video Technology, 1999, 9: 895～908
66 吴伯修. 信息论与编码. 南京：东南大学出版社，1991
67 Hagenauer J. Rate-compatible punctured convolutional codes (RCPC codes) and their applications. IEEE Trans Communications, 1988, 36(4): 389～400
68 Albanese A, Biomer J, Edmonds J, Luby M. Priority encoding transmission. IEEE Trans on Information Theory, 1996, 42(6): 1737～1744
69 Liu H, Ma H, Zarki M E. Error control scheme for networks: An overview. MONET-Mobile Networks and Applications, 1997, 2(2): 167～182
70 Eisenberg Y, Luna C E, et al. Joint source coding and transmission power management for energy efficient wireless video communications. IEEE Trans on Circuits and Systems for Video Technology, 2006, 12(6): 411～424
71 Masnick B, Wolf J K. On linear unequal error protection codes. IEEE Trans Information, 1967, 3: 600～607
72 Wu X, Cheng S, Xiong Z. On packetization of embedded multimedia bitstreams. IEEE Trans Multimedia, 2001, 3(1): 132～140
73 Cai H, Shen G, Xiong Z, et al. An optimal packetization scheme for fine granularity scalable bitstream. IEEE International Symposium on Circuits and Systems, 2002, 5: 641～644
74 Cai H, Shen G, Li S, et al. A novel low-complexity packetization method for fine-granularity scalable (FGS) video streaming. In: Proc of the Fourth IEEE Pacific-Rim Conference on Multimedia, 2003

75 Cai H, Zeng B, Shen G, et al. Error-resilient unequal protection of fine granularity scalable video bitstreams. IEEE International Conference on Communication, 2004, 3: 1303～1307

76 Chou P A, Miao Z. Rate-distortion optimized streaming of packetized media. IEEE Trans on Multimedia, 2006, 8(2): 390～404

77 Qu Q, Pei Y, Tian X, et al. Motion-based interactive video coding and delivery over wireless IP networks. IEEE International Conference on Communications, 2005, 2: 1195～1199

78 Qu Q, Pei Y, Modestino J. Robust H. 264 video coding and transmission over bursty packet loss wireless networks. In: Proc of IEEE VTC2003, 2003

79 Worrall S, Sadka A, Sweeney P, et al. Optimal packetisation of MPEG-4 using RTP over mobile networks. IEE Proc Communications, 2001, 148(4): 197～201

80 Ahmad Z, Worrall S, Sadka A, et al. A novel packetisation scheme for MPEG-4 over 3G wireless systems. In: Proc of the Fifth IEE International Conference on 3G Mobile Communication Technologies, 2004. 302～306

81 Zhu B, Yang Y, Chen C, et al. Optimal packetization of fine granularity scalability codestreams for error-prone channels. IEEE International Conference on Image Processing, 2: 185～188

82 Larzon L, Pink S, Fairhurst G. The Lightweight User Datagram Protocol (UDP-Lite). RFC 3828, 2004

83 Casner S, Jacobson V. Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. RFC 2508, 1999

84 Simpson W. IP in IP Tunneling. RFC 1853, 1995

85 Seo M K, Jeong Y W, Kim J K, et al. A new packet loss-resilient duplicated video transmission. In: Proc 2005 Asia-Pacific Conference on Communications, 2005. 1063～1067

86 Ke C, Lin C, Shieh C, et al. A novel realistic simulation tool for video transmission over wireless network. The IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006

75 Cai H, Zeng B, Shen G, et al. Error-resilient unequal protection of fine granularity scalable video bitstreams. IEEE International Conference on Communication, 2004, 3: 1303～1307
76 Chou P A, Miao Z. Rate-distortion optimized streaming of packetized media. IEEE Trans on Multimedia, 2006, 8(2): 390～404
77 Qu Q, Pei Y, Tian X, et al. Motion-based interactive video coding and delivery over wireless IP networks. IEEE International Conference on Communications, 2005, 2: 1195～1199
78 Qu Q, Pei Y, Modestino J. Robust H. 264 video coding and transmission over bursty packet loss wireless networks. In: Proc of IEEE VTC2003, 2003
79 Worrall S, Sadka A, Sweeney P, et al. Optimal packetisation of MPEG-4 using RTP over mobile networks. IEE Proc Communications, 2001, 148(4): 197～201
80 Ahmad Z, Worrall S, Sadka A, et al. A novel packetisation scheme for MPEG-4 over 3G wireless systems. In: Proc of the Fifth IEE International Conference on 3G Mobile Communication Technologies, 2004. 302～306
81 Zhu B, Yang Y, Chen C, et al. Optimal packetization of fine granularity scalability codestreams for error-prone channels. IEEE International Conference on Image Processing, 2: 185～188
82 Larzon L, Pink S, Fairhurst G. The Lightweight User Datagram Protocol (UDP-Lite). RFC 3828, 2004
83 Casner S, Jacobson V. Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. RFC 2508, 1999
84 Simpson W. IP in IP Tunneling. RFC 1853, 1995
85 Seo M K, Jeong Y W, Kim J K, et al. A new packet loss-resilient duplicated video transmission. In: Proc 2005 Asia-Pacific Conference on Communications, 2005. 1063～1067
86 Ke C, Lin C, Shieh C, et al. A novel realistic simulation tool for video transmission over wireless network. The IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006

75 Cai H, Zeng B, Shen G, et al. Error-resilient unequal protection of fine granularity scalable video bitstreams. IEEE International Conference on Communication, 2004, 3: 1303～1307
76 Chou P A, Miao Z. Rate-distortion optimized streaming of packetized media. IEEE Trans on Multimedia, 2006, 8(2): 390～404
77 Qu Q, Pei Y, Tian X, et al. Motion-based interactive video coding and delivery over wireless IP networks. IEEE International Conference on Communications, 2005, 2: 1195～1199
78 Qu Q, Pei Y, Modestino J. Robust H. 264 video coding and transmission over bursty packet loss wireless networks. In: Proc of IEEE VTC2003, 2003
79 Worrall S, Sadka A, Sweeney P, et al. Optimal packetisation of MPEG-4 using RTP over mobile networks. IEE Proc Communications, 2001, 148(4): 197～201
80 Ahmad Z, Worrall S, Sadka A, et al. A novel packetisation scheme for MPEG-4 over 3G wireless systems. In: Proc of the Fifth IEE International Conference on 3G Mobile Communication Technologies, 2004. 302～306
81 Zhu B, Yang Y, Chen C, et al. Optimal packetization of fine granularity scalability codestreams for error-prone channels. IEEE International Conference on Image Processing, 2: 185～188
82 Larzon L, Pink S, Fairhurst G. The Lightweight User Datagram Protocol (UDP-Lite). RFC 3828, 2004
83 Casner S, Jacobson V. Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. RFC 2508, 1999
84 Simpson W. IP in IP Tunneling. RFC 1853, 1995
85 Seo M K, Jeong Y W, Kim J K, et al. A new packet loss-resilient duplicated video transmission. In: Proc 2005 Asia-Pacific Conference on Communications, 2005. 1063～1067
86 Ke C, Lin C, Shieh C, et al. A novel realistic simulation tool for video transmission over wireless network. The IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006

75 Cai H, Zeng B, Shen G, et al. Error-resilient unequal protection of fine granularity scalable video bitstreams. IEEE International Conference on Communication, 2004, 3: 1303～1307
76 Chou P A, Miao Z. Rate-distortion optimized streaming of packetized media. IEEE Trans on Multimedia, 2006, 8(2): 390～404
77 Qu Q, Pei Y, Tian X, et al. Motion-based interactive video coding and delivery over wireless IP networks. IEEE International Conference on Communications, 2005, 2: 1195～1199
78 Qu Q, Pei Y, Modestino J. Robust H. 264 video coding and transmission over bursty packet loss wireless networks. In: Proc of IEEE VTC2003, 2003
79 Worrall S, Sadka A, Sweeney P, et al. Optimal packetisation of MPEG-4 using RTP over mobile networks. IEE Proc Communications, 2001, 148(4): 197～201
80 Ahmad Z, Worrall S, Sadka A, et al. A novel packetisation scheme for MPEG-4 over 3G wireless systems. In: Proc of the Fifth IEE International Conference on 3G Mobile Communication Technologies, 2004. 302～306
81 Zhu B, Yang Y, Chen C, et al. Optimal packetization of fine granularity scalability codestreams for error-prone channels. IEEE International Conference on Image Processing, 2: 185～188
82 Larzon L, Pink S, Fairhurst G. The Lightweight User Datagram Protocol (UDP-Lite). RFC 3828, 2004
83 Casner S, Jacobson V. Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. RFC 2508, 1999
84 Simpson W. IP in IP Tunneling. RFC 1853, 1995
85 Seo M K, Jeong Y W, Kim J K, et al. A new packet loss-resilient duplicated video transmission. In: Proc 2005 Asia-Pacific Conference on Communications, 2005. 1063～1067
86 Ke C, Lin C, Shieh C, et al. A novel realistic simulation tool for video transmission over wireless network. The IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006

75 Cai H, Zeng B, Shen G, et al. Error-resilient unequal protection of fine granularity scalable video bitstreams. IEEE International Conference on Communication,2004,3:1303～1307
76 Chou P A, Miao Z. Rate-distortion optimized streaming of packetized media. IEEE Trans on Multimedia,2006,8(2): 390～404
77 Qu Q,Pei Y, Tian X, et al. Motion-based interactive video coding and delivery over wireless IP networks. IEEE International Conference on Communications,2005,2: 1195～1199
78 Qu Q,Pei Y, Modestino J. Robust H. 264 video coding and transmission over bursty packet loss wireless networks. In: Proc of IEEE VTC2003,2003
79 Worrall S, Sadka A, Sweeney P, et al. Optimal packetisation of MPEG-4 using RTP over mobile networks. IEE Proc Communications,2001,148(4): 197～201
80 Ahmad Z, Worrall S, Sadka A, et al. A novel packetisation scheme for MPEG-4 over 3G wireless systems. In: Proc of the Fifth IEE International Conference on 3G Mobile Communication Technologies,2004. 302～306
81 Zhu B,Yang Y,Chen C,et al. Optimal packetization of fine granularity scalability codestreams for error-prone channels. IEEE International Conference on Image Processing,2: 185～188
82 Larzon L,Pink S,Fairhurst G. The Lightweight User Datagram Protocol (UDP-Lite). RFC 3828,2004
83 Casner S,Jacobson V. Compressing IP/UDP/RTP Headers for Low-Speed Serial Links. RFC 2508,1999
84 Simpson W. IP in IP Tunneling. RFC 1853,1995
85 Seo M K,Jeong Y W,Kim J K,et al. A new packet loss-resilient duplicated video transmission. In: Proc 2005 Asia-Pacific Conference on Communications,2005. 1063～1067
86 Ke C,Lin C, Shieh C, et al. A novel realistic simulation tool for video transmission over wireless network. The IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing,2006

英汉对照术语表

A

access control (AC)	访问控制
access control list (ACL)	访问控制表
access router (AR)	访问路由器
access requestor (AR)	访问请求者
accounting request (ACR)	计费请求
accounting answer (ACA)	计费应答
adaptive transmission rate control (ARC)	自适应传输速率控制
attestation identity key (AIK)	身份证明密钥
attribute value pair (AVP)	属性值对
authentication and key agreement (AKA)	认证和密钥协商
asymmetric cryptosystem	非对称密码系统
application layer multicast (ALM)	应用层组播
average luminance value (ALV)	平均亮度值
automatic repeat request (ARQ)	自动请求重发

C

care-of address (CoA)	转交地址
certification authorities	认证机构
challenge-handshake authentication protocol (CHAP)	挑战-握手认证协议
colored Petri net (CPN)	着色 Petri 网
controllability	可控性
correspondent node (CN)	通信节点
credential	信任状
cryptographic message syntax (CMS)	密码消息语法

D

data encryption key (DEK)	数据加密密钥
digital rights management (DRM)	数字权限管理
digital watermarking	数字水印
discrete cosine transform (DCT)	离散余弦变换
discretionary access control (DAC)	自主访问控制
distance vector multicast routing protocol (DVMRP)	距离向量组播路由协议
dual-encryption protocol (DEP)	双重加密协议

dual directional hash chains (DDHC)	双向哈希链
dynamic separation of duties (DSD)	动态职责分离

E

elliptic curve cryptosystem (ECC)	椭圆曲线密码体制
encryption sequence (ES)	加密序列
endorsement key (EK)	背签密钥
error concealment	误码隐藏
extensible authentication protocol(EAP)	扩展认证协议

F

forward error correction (FEC)	前向纠错

G

group controller (GC)	组控制器
group key (GK)	组密钥
group key management protocol (GCP)	组密钥管理协议

H

header extension code (HEC)	头扩展码
hierarchical mobile IPv6 (HMIPv6)	层次移动 IPv6
home agent (HA)	家乡代理;用户归属域代理
home location register (HLR)	归属位置登记器
home public land mobile network (HPLMN)	归属公众陆地移动通信网
home subscriber server (HSS)	归属用户服务器
hybrid error correction (HEC)	混合纠错技术

I

integrity	完整性
intra-domain group key management (IGCP)	域内组密钥管理
inverse quantization (IQ)	逆量化

K

key authentication center (KAC)	可信密钥认证中心
key distribution center (KDC)	密钥分配中心
key encryption key (KEK)	密钥加密密钥

L

least privilege	最小特权
light-weighted ELK(LELK)	轻权 ELK
logical key hierarchy (LKH)	逻辑密钥层次
low pass extrapolation (LPE)	低通扩充

M

mandatory access control (MAC)	强制访问控制
media-dependent secure multicast protocol (MSMP)	媒体相关的安全组播协议
micro-electro-mechanism system (MEMS)	微机电系统
mobile anchor point (MAP)	移动锚点
mobile IPv6 (MIPv6)	移动 IPv6
mobile node (MN)	移动节点
multi-level security (MLS)	多级安全
multimedia broadcast multicast service (MBMS)	多媒体广播和多播服务
multiple description coding (MDC)	多描述编码

N

network access server (NAS)	网络接入服务器
network mobility (NEMO)	移动网络

O

on-link CoA (LCoA)	链路转交地址

P

packet data gate-way (PDG)	分组数据网关
password authentication protocol (PAP)	密码认证协议
peak signal noise ratio (PSNR)	峰值信噪比
policy decision point (PDP)	策略决策点
policy enforcement point (PEP)	策略实施点
pseudo random function (PRF)	伪随机函数
pseudo random generator	伪随机数生成器

Q

quality of service (QoS)	服务质量

R

real-time transport control protocol (RTCP)	实时传输控制协议
real-time transport protocol (RTP)	实时传输协议
real-time streaming protocol (RTSP)	实时流协议
regional care-of address (RCoA)	区域转交地址
remote authentication dial-in user service (RADIUS)	远程认证拨号用户服务
resource reserve protocol (RSVP)	资源预留协议
role-based access control (RBAC)	基于角色的访问控制
root of trust	信任根

S

scalable multicast key distribution (SMKD)	可扩展组通信密钥分发
secure socket layer (SSL)	安全套接层
self-healing group key distribution scheme (S-GKDS)	自愈的组密钥分发协议
separation of duties (SoD)	职责分离
session key (SK)	会话密钥
signal noise ratio (SNR)	信噪比
spread spectrum communication	扩展频谱通信
static separation of duties (SSD)	静态职责分离
stream control transmission protocol (SCTP)	流控制传输协议
survivability	可生存性
symmetric cryptosystem	对称密码
synchronized group key distribution protocol (SGKDP)	同步组密钥分发协议
system on chip (SoC)	片上系统

T

traffic encryption key (TEK)	通信加密密钥
transport layer security protocol (TLS)	安全传输层协议
trusted computing (TC)	可信计算
Trusted Computing Group (TCG)	可信计算组
Trusted Computing Platform Alliance (TCPA)	可信计算平台联盟
trusted network connect (TNC)	可信网络连接
trusted platform module (TPM)	可信平台模块
tunneled transport layer security (TTLS)	隧道传输层安全协议

U

unequal error protection (UEP)	非对等保护
universal mobile telecommunications system (UMTS)	通用移动通信系统
usage control (UCON)	使用控制

V

variable length encoding (VLE)	变长解码
variable length decoding (VLD)	变长编码
video encryption algorithm (VEA)	视频加密算法
visited public land mobile network (VPLMN)	受访公众陆地移动通信网

W

wireless personal networks (WPN)	无线个人网络
wireless sensor networks (WSN)	无线传感器网络
WLAN access gateway (WAG)	WLAN 接入网关